

CALL-BY-VALUE AND CALL-BY-NAME DUAL CALCULI WITH INDUCTIVE AND COINDUCTIVE TYPES *

DAISUKE KIMURA AND MAKOTO TATSUTA

National Institute of Informatics, 2-1-2 Hitotsubashi, Tokyo 101-8430, Japan
e-mail address: {kmr, tatsuta}@nii.ac.jp

ABSTRACT. This paper extends the dual calculus with inductive types and coinductive types. The paper first introduces a non-deterministic dual calculus with inductive and coinductive types. Besides the same duality of the original dual calculus, it has the duality of inductive and coinductive types, that is, the duality of terms and coterms for inductive and coinductive types, and the duality of their reduction rules. Its strong normalization is also proved, which is shown by translating it into a second-order dual calculus. The strong normalization of the second-order dual calculus is proved by translating it into the second-order symmetric lambda calculus. This paper then introduces a call-by-value system and a call-by-name system of the dual calculus with inductive and coinductive types, and shows the duality of call-by-value and call-by-name, their Church-Rosser properties, and their strong normalization. Their strong normalization is proved by translating them into the non-deterministic dual calculus with inductive and coinductive types.

1. INTRODUCTION

Dual Calculus DC given by Wadler [27, 28] is a type system which corresponds to the classical sequent calculus LK (see, for example, [7]). It represents computation induced by cut elimination in LK by using its expressions and their reduction. The dual calculus has two nice properties: computation in classical logic, and duality.

The computation of classical logic has been intensively studied, for example, [2, 4, 8, 9, 20, 21, 23, 27, 28]. They all studied the Curry-Howard correspondence between classical logic and functional programming languages with sophisticated control structures like catch/throw and first-class continuations. This correspondence is an extension of the Curry-Howard correspondence between intuitionistic logic and the typed λ -calculus, which is well established.

The classical sequent calculus LK has nice duality. We have an involution that maps conjunction and disjunction to each other, and maps the left and right rules of conjunction to the right and left rules of disjunction and vice versa. This involution can be extended to the cut elimination procedure for LK.

The system DC inherits the duality of the classical sequent calculus LK. Moreover, its proof terms called terms, coterms, and statements also have duality, since they correspond to proofs in

2012 ACM CCS: [Theory of computation]: Models of computation—Computability—Lambda calculus.

Key words and phrases: Curry-Howard isomorphism, Classical logic, Dual Calculus, Inductive definitions, Coinductive definitions.

* The conference version of this paper has appeared in [13].

LK. This implies that its reduction relation can have duality since the reduction relation is induced by the cut elimination procedure in LK. In this framework, Wadler gave the call-by-value and call-by-name strategies in DC, and showed the duality of them [27]. He also showed that the equational correspondence between DC and Parigot's $\lambda\mu$ -calculus [19], and showed the duality between call-by-value and call-by-name of the $\lambda\mu$ -calculus using the duality of the dual calculus [28]. Since then, the dual calculus has been actively studied [26, 12, 13].

Inductive definitions are important in both mathematical logic and computer science. Inductive definitions strengthen expressiveness of logical systems (for example, See [3]). They are central in programming and program verification [22, 18, 14] for handling recursive data structures such as lists and trees, and specification of recursive programs. Coinductive definitions are also important since they can represent streams, infinite trees, and bisimulation, for example, in [24].

This paper presents Dual Calculus $DC_{\mu\nu}$ with inductive types and coinductive types. Our calculus extends the duality of DC to inductive types and coinductive types. The involution in DC is extended so that it maps inductive types and coinductive types to each other. It also maps the left and right rules of inductive types to the right and left rules of coinductive types and vice versa. Because of the duality of the proof rules, we will have cut elimination procedure that keeps the duality of inductive types and coinductive types. This induces the duality of the reduction relations of proof terms for inductive types and coinductive types.

Our main results are: (1) the duality between inductive types and coinductive types with reduction, (2) strong normalization in $DC_{\mu\nu}$, (3) strong normalization in the second-order Dual Calculus DC2, (4) the duality between the call-by-value and call-by-name $DC_{\mu\nu}$, and (5) the Church-Rosser property and strong normalization of the call-by-value and call-by-name $DC_{\mu\nu}$.

We will show strong normalization of $DC_{\mu\nu}$. In order for proving the strong normalization, we will first show the strong normalization of the second-order Dual Calculus DC2 given by [26] by interpreting it in second-order symmetric lambda-calculus given in [21]. Then strong normalization of $DC_{\mu\nu}$ is proved by interpreting it in DC2 by using second-order coding of inductive and coinductive types.

We first introduce the system $DC_{\mu\nu}$ that does not have reduction strategies, since it is designed by the Curry-Howard correspondence for a standard cut elimination procedure in LK. The system can discuss non-deterministic aspects of computation in classical logic, since the execution of programs in $DC_{\mu\nu}$ is non-deterministic. It also works as a base framework for other variants of $DC_{\mu\nu}$ with specific reduction strategies such as call-by-value and call-by-name that will be given later.

The duality between call-by-value and call-by-name is first suggested by Filinski [5]. The dual calculus gives a clear explanation for this duality by using the logical duality of classical logic. We will show the duality of call-by-value and call-by-name in the dual calculus extended with inductive types and coinductive types. We extend the call-by-value DC and the call-by-name DC given in [27] with inductive types and coinductive types, and introduce the systems CBV $DC_{\mu\nu}$ and CBN $DC_{\mu\nu}$. They are obtained from $DC_{\mu\nu}$ by restricting its non-deterministic reduction to the call-by-value or call-by-name strategies, and also by adding some strategy-specific reduction rules. In the same way as [27], we show the duality of call-by-value and call-by-name in the dual calculus with inductive types and coinductive types. We will show the Church-Rosser property as well as strong normalization for CBV $DC_{\mu\nu}$ and CBN $DC_{\mu\nu}$. The strong normalization will be shown by translating CBV $DC_{\mu\nu}$ and CBN $DC_{\mu\nu}$ into $DC_{\mu\nu}$.

In [1], the duality between inductive types and coinductive types in linear logic is studied. Our system $DC_{\mu\nu}$ shows the duality in ordinary sequent calculus LK.

Momigliano and Tiu [16, 17] discussed an intuitionistic sequent calculus with inductive definitions and coinductive definitions and showed its cut elimination theorem. Our system $DC_{\mu\nu}$ is

a classical system and our strong normalization shows the cut elimination theorem of the classical sequent calculus. Our cut elimination procedure is not closed in an intuitionistic fragment because it keeps the duality and we have the corresponding proof rule that manipulates a succedent if we have some proof rule that manipulates an antecedent. So we cannot directly compare our method and their method.

In category theory, inductive definitions are represented by initial algebras and coinductive definitions are represented by final coalgebras [6], and their duality in category theory is known. Our system $DC\mu\nu$ enables us to show the duality in a clear syntactic way by using a type system.

Several papers for dual calculus investigated the duality of computation. Wadler showed the duality between values and continuations, and the duality between call-by-value computation and call-by-name computation by using the explicit duality of DC [27, 28]. The first author of this paper showed the duality between the call-by-name fixed point operator and the call-by-value loop operator by extending DC [11]. The first author also showed the duality of reduction between call-by-value computation and call-by-name computation in $\lambda\mu$ -calculus by using DC [12] to answer the open question presented in Wadler's invited talk at RTA2005 [28], which asked whether the duality between call-by-value and call-by-name in his equation systems would be refined in reduction systems. Tzevelekos [26] investigated the dual calculus given in [27]. He assumed some additional conditions on reductions, and showed both Church-Rosser property and strong normalization hold under his conditions. He also investigated the relationship between DC and the symmetric λ -calculus by Barbanera and Berardi [2]. A second-order extension of DC is also considered in [26].

The system $\bar{\mu}\tilde{\mu}$ in [4] is a system with implication and subtraction, and also has duality. Their calculus with negation, conjunction, and disjunction is called $\mu\tilde{\mu}^{\wedge_a \vee_a \neg}$ and the correspondence between it and the dual calculus is discussed in [9].

A semantical approach to the duality between call-by-value and call-by-name was studied by Selinger [23]. He gave a categorical semantics of the $\lambda\mu$ -calculus, and explained the duality by using the categorical duality. This approach is extended to the duality between the fixed point operator and the loop operator by Kakutani [10].

Section 2 gives a definition of DC and states its duality. Section 3 introduces $DC\mu\nu$ and shows its duality. Section 4 gives examples. In section 5, we give DC2 and show its strong normalization. Section 6 proves strong normalization for $DC\mu\nu$. Section 7 introduces CBV $DC\mu\nu$ and CBN $DC\mu\nu$ and shows their Church-Rosser properties and strong normalization.

2. THE DUAL CALCULUS DC

This section defines Dual Calculus DC and states its duality. This system is obtained from the original Dual Calculus given in [27] by removing reduction strategies in reduction rules. This system gives us a base framework for several variants of dual calculi.

Definition 2.1 (Types and Expressions of DC). Let X, Y, Z, \dots range over type variables, A, B, \dots range over types, The symbols x, y, z, \dots range over variables, and $\alpha, \beta, \gamma, \dots$ range over covariables. We assume an involution $(-)'$ between variables and covariables, which satisfies $x'' = x$ and $\alpha'' = \alpha$. An expression (denoted by D, E, \dots) is either a term (denoted by M, N, \dots), a coterms (denoted by K, L, \dots), or a statement (denoted by S, T, \dots). We define them as follows:

Types	$A ::= X \mid A \wedge A \mid A \vee A \mid \neg A,$
Expressions	$D ::= M \mid K \mid S,$
Terms	$M ::= x \mid \langle M, M \rangle \mid \langle M \rangle \text{inl} \mid \langle M \rangle \text{inr} \mid [K] \text{not} \mid (S). \alpha,$
Coterms	$K ::= \alpha \mid [K, K] \mid \text{fst}[K] \mid \text{snd}[K] \mid \text{not} \langle M \rangle \mid x.(S),$
Statements	$S ::= M \bullet K.$

The term $(S).\alpha$ binds the covariable α in S . The coterms $x.(S)$ binds the variable x in S . We write $FV(D)$ for the set of free variables in D . We also write $FCV(D)$ for the set of free covariables in D . We will use $[_\cdot/_\cdot]$ for substitution. For example, the substitution $S[M/x]$ denotes the statement obtained from S by replacing x by M .

The type $A \wedge B$ denotes a conjunction, $A \vee B$ denotes a disjunction, and $\neg A$ denotes a negation. A variable means an ordinary variable. A covariable means an output port and gets some value after computation. A term represents an ordinary computation which becomes a value or puts values at output ports after computation. The term $\langle M, N \rangle$ means a pair. The terms $\langle M \rangle \text{inl}$ and $\langle M \rangle \text{inr}$ mean the left injection and the right injection to a disjoint sum, respectively. When $[K] \text{not}$ gets its input, it gives the input to K and computes K . The term $(S).\alpha$ is an abstraction of S by α . It computes S and its value is the value at the output port α . A coterms represents continuation which puts values at output ports after computation when it gets its input. The coterms $[K, L]$ gets an input of a disjoint sum. If the input is $\langle M \rangle \text{inl}$, it gives M to K and computes K . If the input is $\langle M \rangle \text{inr}$, it gives M to L and computes L . The coterms $\text{fst}[K]$ gets an input of a pair. If the input is $\langle M, N \rangle$, then it gives M to K and computes K . The coterms $\text{snd}[K]$ also gets an input of a pair. If the input is $\langle M, N \rangle$, then it gives N to K and computes K . The coterms $\text{not}\langle M \rangle$ gets a continuation as its input. It gives M to the continuation and computes the continuation. The coterms $x.(S)$ is an abstraction of S by x . If it gets an input, it puts the input in x and computes S . The statement $M \bullet K$ means the computation of K with the input M that may put values at output ports.

A typing judgment (denoted by J) of DC takes either the form $\Gamma \vdash \Delta \mid M : A$, the form $K : A \mid \Gamma \vdash \Delta$, or the form $\Gamma \mid S \vdash \Delta$, where Γ denotes a context $x_1 : A_1, \dots, x_n : A_n$ that is a set of variable declarations, and Δ denotes a cocontext $\alpha_1 : B_1, \dots, \alpha_m : B_m$ that is a set of covariable declarations. We will call M , K , and S a principal expression in those judgments. The domain of Γ (denoted by $\text{dom}(\Gamma)$) is the set of variables $\{x_1, \dots, x_n\}$ if Γ is $x_1 : A_1, \dots, x_n : A_n$. The domain of Δ (denoted by $\text{dom}(\Delta)$) is the set of covariables $\{\alpha_1, \dots, \alpha_m\}$ if Δ is $\alpha_1 : B_1, \dots, \alpha_m : B_m$.

We intuitively explain the typing judgments. There can be other ways of intuitive explanation, for example, [26]. In order to give an intuitive idea in general, we assume an evaluation strategy for expressions, and a notion of values for the strategy. For example, when we take call-by-name, the values will be canonical form, and the computation will be lazy evaluation. The focus \mid is used only for denoting which part contains a term, a coterms, or a statement in a judgment, and when we think the corresponding sequent in ordinary sequent calculus, we will erase it. The typing judgment $x_1 : A_1, \dots, x_n : A_n \vdash \alpha_1 : B_1, \dots, \alpha_m : B_m \mid M : A$ means that when each x_i has a value of type A_i , and M is computed, then M returns a value of type A or some α_i gets a value of type B_i . The judgment $K : A \mid x_1 : A_1, \dots, x_n : A_n \vdash \alpha_1 : B_1, \dots, \alpha_m : B_m$ means that when each x_i has a value of type A_i , an input of type A is given to K , and K is computed, then some α_i gets a value of type B_i . The judgment $x_1 : A_1, \dots, x_n : A_n \mid S \vdash \alpha_1 : B_1, \dots, \alpha_m : B_m$ means that when each x_i has a value of type A_i and S is computed, then some α_i gets a value of type B_i . We sometimes use the symbol \vdash_{DC} instead of the symbol \vdash that appears in a judgment in order to explicitly show it is a judgment of DC. That is, we write $\Gamma \vdash_{\text{DC}} \Delta \mid M : A$ for the judgment $\Gamma \vdash \Delta \mid M : A$. Similarly, we write $K : A \mid \Gamma \vdash_{\text{DC}} \Delta$ and $\Gamma \mid S \vdash_{\text{DC}} \Delta$.

The typing rules are given in Figure 1. If we erase terms, coterms, statements, and the symbol \mid , the system becomes logically equivalent to a fragment of classical sequent calculus LK, whose definition is given in, for example, [7].

$$\begin{array}{c}
\frac{}{\Gamma, x : A \vdash \Delta \mid x : A} (AxR) \qquad \frac{}{\alpha : A \mid \Gamma \vdash \Delta, \alpha : A} (AxL) \\
\frac{\Gamma \vdash \Delta \mid M : A \quad \Gamma \vdash \Delta \mid N : B}{\Gamma \vdash \Delta \mid \langle M, N \rangle : A \wedge B} (\wedge R) \qquad \frac{K : A \mid \Gamma \vdash \Delta \quad L : B \mid \Gamma \vdash \Delta}{[K, L] : A \vee B \mid \Gamma \vdash \Delta} (\vee L) \\
\frac{\Gamma \vdash \Delta \mid M : A}{\Gamma \vdash \Delta \mid \langle M \rangle \text{inl} : A \vee B} (\vee R1) \qquad \frac{K : A \mid \Gamma \vdash \Delta}{\text{fst}[K] : A \wedge B \mid \Gamma \vdash \Delta} (\wedge L1) \\
\frac{\Gamma \vdash \Delta \mid M : B}{\Gamma \vdash \Delta \mid \langle M \rangle \text{inr} : A \vee B} (\vee R2) \qquad \frac{K : B \mid \Gamma \vdash \Delta}{\text{snd}[K] : A \wedge B \mid \Gamma \vdash \Delta} (\wedge L2) \\
\frac{K : A \mid \Gamma \vdash \Delta}{\Gamma \vdash \Delta \mid [K] \text{not} : \neg A} (\neg R) \qquad \frac{\Gamma \vdash \Delta \mid M : A}{\text{not} \langle M \rangle : \neg A \mid \Gamma \vdash \Delta} (\neg L) \\
\frac{\Gamma \mid S \vdash \Delta, \alpha : A}{\Gamma \vdash \Delta \mid (S). \alpha : A} (IR) \qquad \frac{\Gamma, x : A \mid S \vdash \Delta}{x.(S) : A \mid \Gamma \vdash \Delta} (IL) \\
\frac{\Gamma \vdash \Delta \mid M : A \quad K : A \mid \Gamma \vdash \Delta}{\Gamma \mid M \bullet K \vdash \Delta} (Cut)
\end{array}$$

Figure 1: Typing rules of DC

Definition 2.2 (Reduction). The reduction relation $\longrightarrow_{\text{DC}}$ is defined as the compatible closure of the following reduction rules:

$$\begin{array}{ll}
(\beta \wedge_1) & \langle M, N \rangle \bullet \text{fst}[K] \longrightarrow_{\text{DC}} M \bullet K, \\
(\beta \wedge_2) & \langle M, N \rangle \bullet \text{snd}[K] \longrightarrow_{\text{DC}} N \bullet K, \\
(\beta \vee_1) & \langle M \rangle \text{inl} \bullet [K, L] \longrightarrow_{\text{DC}} M \bullet K, \\
(\beta \vee_2) & \langle M \rangle \text{inr} \bullet [K, L] \longrightarrow_{\text{DC}} M \bullet L, \\
(\beta \neg) & [K] \text{not} \bullet \text{not} \langle M \rangle \longrightarrow_{\text{DC}} M \bullet K, \\
(\beta R) & (S). \alpha \bullet K \longrightarrow_{\text{DC}} S [K/\alpha], \\
(\beta L) & M \bullet x.(S) \longrightarrow_{\text{DC}} S [M/x], \\
(\eta R) & (M \bullet \alpha). \alpha \longrightarrow_{\text{DC}} M, \\
(\eta L) & x.(x \bullet K) \longrightarrow_{\text{DC}} K,
\end{array}$$

where x and α are fresh in (ηL) and (ηR) , respectively.

The rules (ηR) and (ηL) are necessary to get the results of computation of terms and coterms from computation of statements inside them. We do not include the η -rules for logical connectives that are given in [28], since these break the confluence property for call-by-value and call-by-name systems, which we will study in Section 7. In order to study a base framework, we first consider a non-deterministic rewriting system that does not commit to either the call-by-name or call-by-value theory.

The system DC we consider first is obtained from the original dual calculus given in [27] by omitting evaluation strategies, dropping (ζ) -rules that provide strong evaluation under call-by-value and call-by-name strategies, and replacing (ηL) and (ηR) -expansion rules by (ηL) and (ηR) -reduction rules.

The role of (ηL) and (ηR) -reduction rules are to simplify logical proofs without changing any proof structure. In the last section, we also give the call-by-value and call-by-name variants of $\text{DC}_{\mu\nu}$. The role of these rules become clearer in that section since they are necessary to obtain a value as the result of a computation under some strategy.

The type of an expression is preserved by reduction.

Proposition 2.3 (Subject reduction of DC). *The following claims hold.*

- (1) *If $\Gamma \vdash_{\text{DC}} \Delta \mid M : A$ and $M \longrightarrow_{\text{DC}} N$, then $\Gamma \vdash_{\text{DC}} \Delta \mid N : A$ holds.*
- (2) *If $K : A \mid \Gamma \vdash_{\text{DC}} \Delta$ and $K \longrightarrow_{\text{DC}} L$, then $L : A \mid \Gamma \vdash_{\text{DC}} \Delta$ holds.*
- (3) *If $\Gamma \mid S \vdash_{\text{DC}} \Delta$ and $S \longrightarrow_{\text{DC}} T$, then $\Gamma \mid T \vdash_{\text{DC}} \Delta$ holds.*

This proposition is shown by induction on reduction using the following substitution lemma.

Lemma 2.4 (Substitution lemma). *The following claims hold.*

- (1) *Suppose $\Gamma \vdash_{\text{DC}} \Delta \mid N : A$ is derivable. Then we have the following.*
 - (1a) *If $\Gamma, x : A \vdash_{\text{DC}} \Delta \mid M : B$, then $\Gamma \vdash_{\text{DC}} \Delta \mid M[N/x] : B$,*
 - (1b) *if $K : B \mid \Gamma, x : A \vdash_{\text{DC}} \Delta$, then $K[N/x] : B \mid \Gamma \vdash_{\text{DC}} \Delta$, and*
 - (1c) *if $\Gamma, x : A \mid S \vdash_{\text{DC}} \Delta$, then $\Gamma \mid S[N/x] \vdash_{\text{DC}} \Delta$.*
- (2) *Suppose $L : A \mid \Gamma \vdash_{\text{DC}} \Delta$ is derivable. Then we have the following.*
 - (2a) *If $\Gamma \vdash_{\text{DC}} \Delta, \alpha : A \mid M : B$, then $\Gamma \vdash_{\text{DC}} \Delta \mid M[L/\alpha] : B$,*
 - (2b) *if $K : B \mid \Gamma \vdash_{\text{DC}} \Delta, \alpha : A$, then $K[L/\alpha] : B \mid \Gamma \vdash_{\text{DC}} \Delta$, and*
 - (2c) *if $\Gamma \mid S \vdash_{\text{DC}} \Delta, \alpha : A$, then $\Gamma \mid S[L/\alpha] \vdash_{\text{DC}} \Delta$.*

Proof. The claims (1a), (1b), and (1c) are shown simultaneously by induction on M , K , and S . The claims (2a), (2b), and (2c) are also shown simultaneously by induction on M , K , and S . \square

The following duality transformation extends the duality in the sequent calculus LK to terms, coterms, and statements.

Definition 2.5 (Duality Transformation). The duality transformation $(-)^{\circ}$ from DC into itself is defined for types and expressions as follows:

$$\begin{aligned}
(X)^{\circ} &= X, & (\neg A)^{\circ} &= \neg(A)^{\circ}, & (A \wedge B)^{\circ} &= (A)^{\circ} \vee (B)^{\circ}, & (A \vee B)^{\circ} &= (A)^{\circ} \wedge (B)^{\circ}, \\
(x)^{\circ} &= x', & (\alpha)^{\circ} &= \alpha', \\
\langle\langle M, N \rangle\rangle^{\circ} &= [\langle M \rangle^{\circ}, \langle N \rangle^{\circ}], & [\langle K, L \rangle]^{\circ} &= \langle\langle K \rangle^{\circ}, \langle L \rangle^{\circ}\rangle, \\
\langle\langle M \rangle^{\text{inl}}\rangle^{\circ} &= \text{fst}[\langle M \rangle^{\circ}], & \langle\text{fst}[K]\rangle^{\circ} &= \langle\langle K \rangle^{\circ}\rangle^{\text{inl}}, \\
\langle\langle M \rangle^{\text{inr}}\rangle^{\circ} &= \text{snd}[\langle M \rangle^{\circ}], & \langle\text{snd}[K]\rangle^{\circ} &= \langle\langle K \rangle^{\circ}\rangle^{\text{inr}}, \\
[\langle K \rangle^{\text{not}}]^{\circ} &= \text{not}\langle\langle K \rangle^{\circ}\rangle, & \langle\text{not}\langle M \rangle^{\circ}\rangle^{\circ} &= [\langle M \rangle^{\circ}]^{\text{not}}, \\
\langle\langle S \rangle^{\circ}.\alpha\rangle^{\circ} &= \alpha'.\langle\langle S \rangle^{\circ}\rangle, & \langle x.\langle S \rangle^{\circ}\rangle^{\circ} &= \langle\langle S \rangle^{\circ}\rangle.x', \\
(M \bullet K)^{\circ} &= (K)^{\circ} \bullet (M)^{\circ}.
\end{aligned}$$

Note that a type and a statement are mapped to themselves. A term and a coterms are mapped to each other.

We also define transformation for judgments. If Γ is $x_1 : A_1, \dots, x_n : A_n$, then $(\Gamma)^{\circ}$ is defined as $(x_1)^{\circ} : (A_1)^{\circ}, \dots, (x_n)^{\circ} : (A_n)^{\circ}$. If Δ is $\alpha_1 : B_1, \dots, \alpha_m : B_m$, then $(\Delta)^{\circ}$ is defined as $(\alpha_1)^{\circ} : (B_1)^{\circ}, \dots, (\alpha_m)^{\circ} : (B_m)^{\circ}$. The judgment $(\Gamma \vdash \Delta \mid M : A)^{\circ}$ is defined as $(M)^{\circ} : (A)^{\circ} \mid (\Delta)^{\circ} \vdash (\Gamma)^{\circ}$. The judgment $(K : A \mid \Gamma \vdash \Delta)^{\circ}$ is defined as $(\Delta)^{\circ} \vdash (\Gamma)^{\circ} \mid (K)^{\circ} : (A)^{\circ}$. The judgment $(\Gamma \mid S \vdash \Delta)^{\circ}$ is defined as $(\Delta)^{\circ} \mid (S)^{\circ} \vdash (\Gamma)^{\circ}$.

We also define transformation for inference rule names as follows: $(AxR)^{\circ} = (AxL)$, $(AxL)^{\circ} = (AxR)$, $(\vee R1)^{\circ} = (\wedge L1)$, $(\wedge L1)^{\circ} = (\vee R1)$, $(\wedge R)^{\circ} = (\vee L)$, $(\vee L)^{\circ} = (\wedge R)$, $(\vee L2)^{\circ} = (\wedge R2)$, $(\vee R2)^{\circ} = (\wedge L2)$, $(\neg L)^{\circ} = (\neg R)$, $(\neg R)^{\circ} = (\neg L)$, $(IR)^{\circ} = (IL)$, $(IL)^{\circ} = (IR)$, and $(Cut)^{\circ} = (Cut)$.

This duality transformation preserves substitution of terms and coterms.

Lemma 2.6. *The following claims hold.*

- (1) $(D[M/x])^\circ = (D)^\circ[(M)^\circ/x']$.
- (2) $(D[K/\alpha])^\circ = (D)^\circ[(K)^\circ/\alpha']$.

Proof. The claims (1) and (2) are shown by induction on D . We treat the first case of (1): the case of D is x . $(x[M/x])^\circ = (M)^\circ = x'[(M)^\circ/x'] = (x)^\circ[(M)^\circ/x']$. The other cases are straightforwardly proved by the induction hypothesis. \square

This duality transformation is shown to preserve typing and reduction, and to be an involution. This transformation is a homomorphism for this system in the sense that it preserves typing and reduction. An important feature of DC is its duality by this transformation. A term is dual to a cotermin by this homomorphism.

Proposition 2.7 (Duality of DC). *The followings hold.*

- (1) *If J is derived from J_1, \dots, J_n ($n = 1$ or 2) by an inference rule R , then $(J)^\circ$ is derived from $(J_n)^\circ, \dots, (J_1)^\circ$ by the inference rule $(R)^\circ$.*
- (2) $D \longrightarrow_{\text{DC}} E$ implies $(D)^\circ \longrightarrow_{\text{DC}} (E)^\circ$.
- (3) $((A)^\circ)^\circ = A$, $((D)^\circ)^\circ = D$, and $((J)^\circ)^\circ = J$ hold.

Proof. The claim (1) is proved by case analysis of the inference rules. The claim (2) is proved by induction on the generation of $\longrightarrow_{\text{DC}}$ using Lemma 2.6. The claim (3) is proved by induction on types and expressions. \square

Remark 2.8. The $(-)^{\circ}$ transformation maps dual reduction rules to each other. That is, if $D \longrightarrow_{\text{DC}} E$ is the reduction rules $(\beta\wedge_1)$, $(\beta\wedge_2)$, $(\beta\vee_1)$, $(\beta\vee_2)$, $(\beta\neg)$, (βR) , (βL) , (ηR) , and (ηL) , then $(D)^\circ \longrightarrow_{\text{DC}} (E)^\circ$ is the reduction rules $(\beta\vee_1)$, $(\beta\vee_2)$, $(\beta\wedge_1)$, $(\beta\wedge_2)$, $(\beta\neg)$, (βL) , (βR) , (ηL) , and (ηR) , respectively.

Implication \supset can be defined by \neg and \vee in the same way as [27].

Definition 2.9. We write $A \supset B$ for $\neg A \vee B$. We also write $\lambda x.M$ for $(\langle [x.(\langle M \rangle \text{inr} \bullet \gamma)] \text{not} \rangle \text{inl} \bullet \gamma). \gamma$. We also write $N @ K$ for $[\text{not} \langle N \rangle, K]$.

The constructor $@$ simulates the application in λ -calculus together with \bullet . The following holds from the definition.

Proposition 2.10. *The following typing inference rules and reduction rule are derivable.*

$$\frac{\Gamma, x : A \vdash \Delta \mid M : B}{\Gamma \vdash \Delta \mid \lambda x.M : A \supset B} (\supset R) \quad \frac{\Gamma \vdash \Delta \mid M : A \quad K : B \mid \Gamma \vdash \Delta}{M @ K : A \supset B \mid \Gamma \vdash \Delta} (\supset L)$$

$$(\beta \supset) \quad \lambda x.M \bullet (N @ K) \longrightarrow_{\text{DC}} M[N/x] \bullet K$$

3. THE DUAL CALCULUS $\text{DC}_{\mu\nu}$ WITH INDUCTIVE AND COINDUCTIVE TYPES

In this section, we present $\text{DC}_{\mu\nu}$, which is an extension of DC with inductive types and coinductive types. We first extend the definition of types of DC to inductive types $\mu X.A$ and coinductive types $\nu X.A$, and then extend expressions and reduction.

In Section 5, we will introduce the second-order system DC2. The system $\text{DC}_{\mu\nu}$ is worth to be studied as well as DC2, since $\text{DC}_{\mu\nu}$ is within a first-order logic.

We first define types, their positive type variables, and their negative type variables. A positive type variable in a type does not occur negatively in the type in the usual sense. A negative type variable in a type does not occur positively in the type.

Definition 3.1. The set of type variables is written by TyVars . We define the types of $\text{DC}\mu\nu$ (denoted by A, B, \dots) and the set $\text{Pos}(A)$ of *positive type variables* in the type A and the set $\text{Neg}(A)$ of *negative type variables* in the type A as follows:

$$A ::= X \mid A \wedge A \mid A \vee A \mid \neg A \mid \mu X.A \mid \nu X.A$$

where $\mu X.A$ and $\nu X.A$ are defined when the type variable X is in $\text{Pos}(A)$.

$$\begin{aligned} \text{Pos}(X) &= \text{TyVars}, \\ \text{Neg}(X) &= \text{TyVars} \setminus \{X\}, \\ \text{Pos}(A_1 \wedge A_2) &= \text{Pos}(A_1 \vee A_2) = \text{Pos}(A_1) \cap \text{Pos}(A_2), \\ \text{Neg}(A_1 \wedge A_2) &= \text{Neg}(A_1 \vee A_2) = \text{Neg}(A_1) \cap \text{Neg}(A_2), \\ \text{Pos}(\neg B) &= \text{Neg}(B), \\ \text{Neg}(\neg B) &= \text{Pos}(B), \\ \text{Pos}(\mu X.B) &= \text{Pos}(\nu X.B) = \text{Pos}(B) \cup \{X\}, \\ \text{Neg}(\mu X.B) &= \text{Neg}(\nu X.B) = \text{Neg}(B) \cup \{X\}. \end{aligned}$$

The types $\mu X.A$ and $\nu X.A$ bind X in A .

When we think standard semantics of the propositional logic with inductive and coinductive definitions, $\mu X.A$ and $\nu X.A$ are interpreted by the least fixed point and the greatest fixed point of the monotone function \mathcal{P} respectively, where \mathcal{P} is the function which maps a set U to the set $A[U/X]$. Let μ be $\mu X.A$ and ν be $\nu X.A$. They will have the following properties: (a) $A[\mu/X] \subseteq \mu$, (b) $A[B/X] \subseteq B$ implies $\mu \subseteq B$, (c) $\nu \subseteq A[\nu/X]$, and (d) $B \subseteq A[B/X]$ implies $B \subseteq \nu$. Based on this meaning, we will introduce terms, coterms, and their reduction for inductive and coinductive types in the same way as [15].

Definition 3.2. The terms, coterms, and statements of $\text{DC}\mu\nu$ are defined as follows:

$$\begin{aligned} M &::= x \mid \langle M, M \rangle \mid \langle M \rangle \text{inl} \mid \langle M \rangle \text{inr} \mid [K] \text{not} \mid (S). \alpha \mid \text{in}^{\mu X.A} \langle M \rangle \mid \text{coitr}_x^A \langle M, M \rangle, \\ K &::= \alpha \mid [K, K] \mid \text{fst}[K] \mid \text{snd}[K] \mid \text{not} \langle M \rangle \mid x.(S) \mid \text{out}^{\nu X.A} [K] \mid \text{itr}_\alpha^A [K, K], \\ S &::= M \bullet K. \end{aligned}$$

The term $\text{itr}_\alpha^A [K, L]$ binds α in K . The coterm $\text{coitr}_x^A \langle M, N \rangle$ binds x in M .

The expressions $\text{in}^{\mu X.A} \langle M \rangle$ and $\text{itr}_\alpha^A [K, L]$ are the expressions for inductive types. The constructor $\text{in}^{\mu X.A}$ maps a term of type $A[\mu X.A/X]$ to that of $\mu X.A$. The coterm $\text{itr}_\alpha^A [K, L]$ is an iterator having an input of type $\mu X.A$ where L is a postprocessor after iteration. When it gets the input of type $\mu X.A$, first a value of type $A[\mu X.A/X]$ is computed according to the input, next a value of type $A[B/X]$ is computed by recursive invocation of the iterator, then it is given to K and K is computed to get a value of type B , and finally the value is given to L and L is computed. Dually, $\text{out}^{\nu X.A} [K]$ and $\text{coitr}_x^A \langle M, N \rangle$ are defined for coinductive types. The constructor $\text{out}^{\nu X.A}$ maps a coterm of type $\nu X.A$ to that of $A[\nu X.A/X]$. When the coterm $\text{out}^{\nu X.A} [K]$ gets the input of type $\nu X.A$, first the input is transformed into a value of type $A[\nu X.A/X]$, then the value is given to K , and finally K is computed. The term $\text{coitr}_x^A \langle M, N \rangle$ is a coiterator of type $\nu X.A$. It transforms N of type B into a value of $\nu X.A$ according to M . Type annotations will be necessary for defining reduction rules.

Definition 3.3. The typing rules of $\text{DC}\mu\nu$ are defined by those of DC and the following rules:

$$\frac{\Gamma \vdash \Delta \mid M : A[\mu X.A/X]}{\Gamma \vdash \Delta \mid \text{in}^{\mu X.A} \langle M \rangle : \mu X.A} (\mu R) \quad \frac{K : A[B/X] \mid \Gamma \vdash \Delta, \alpha : B \quad L : B \mid \Gamma \vdash \Delta}{\text{itr}_\alpha^A [K, L] : \mu X.A \mid \Gamma \vdash \Delta} (\mu L)$$

$$\frac{K : A[\nu X.A/X] \mid \Gamma \vdash \Delta}{\text{out}^{\nu X.A}[K] : \nu X.A \mid \Gamma \vdash \Delta} (\nu L) \quad \frac{\Gamma, x : B \vdash \Delta \mid M : A[B/X] \quad \Gamma \vdash \Delta \mid N : B}{\Gamma \vdash \Delta \mid \text{coitr}_x^B\langle M, N \rangle : \nu X.A} (\nu R)$$

We sometimes use the symbol $\vdash_{\text{DC}\mu\nu}$ instead of the symbol \vdash in a judgment in order to explicitly show it is a judgment of $\text{DC}\mu\nu$. That is, we write $\Gamma \vdash_{\text{DC}\mu\nu} \Delta \mid M : A$ for the judgment $\Gamma \vdash \Delta \mid M : A$. Similarly, we write $K : A \mid \Gamma \vdash_{\text{DC}\mu\nu} \Delta$ and $\Gamma \mid S \vdash_{\text{DC}\mu\nu} \Delta$.

The system $\text{DC}\mu\nu$ satisfies the following basic lemmas.

Lemma 3.4 (Weakening lemma of $\text{DC}\mu\nu$). Let $\Gamma \subseteq \Gamma'$ and $\Delta \subseteq \Delta'$. Then

- (1) if $\Gamma \vdash_{\text{DC}\mu\nu} \Delta \mid M : A$ is provable, then $\Gamma' \vdash_{\text{DC}\mu\nu} \Delta' \mid M : A$ holds,
- (2) if $K : A \mid \Gamma \vdash_{\text{DC}\mu\nu} \Delta$ is provable, then $K : A \mid \Gamma' \vdash_{\text{DC}\mu\nu} \Delta'$ holds, and
- (3) if $\Gamma \mid S \vdash_{\text{DC}\mu\nu} \Delta$ is provable, then $\Gamma' \mid S \vdash_{\text{DC}\mu\nu} \Delta'$ holds.

Proof. They are shown simultaneously by induction on M , K , and S . □

Lemma 3.5. Let $\Gamma' \subseteq \Gamma$ and $\Delta' \subseteq \Delta$. Then the following claims hold in $\text{DC}\mu\nu$.

- (1) If $\text{FV}(M) \subseteq \text{dom}(\Gamma')$ and $\text{FCV}(M) \subseteq \text{dom}(\Delta')$, then $\Gamma \vdash \Delta \mid M : A$ implies $\Gamma' \vdash \Delta' \mid M : A$.
- (2) If $\text{FV}(K) \subseteq \text{dom}(\Gamma')$ and $\text{FCV}(K) \subseteq \text{dom}(\Delta')$, then $K : A \mid \Gamma \vdash \Delta$ implies $K : A \mid \Gamma' \vdash \Delta'$.
- (3) If $\text{FV}(S) \subseteq \text{dom}(\Gamma')$ and $\text{FCV}(S) \subseteq \text{dom}(\Delta')$, then $\Gamma \mid S \vdash \Delta$ implies $\Gamma' \mid S \vdash \Delta'$.

Proof. They are shown simultaneously by induction on M , K , and S . □

Lemma 3.6 (Substitution lemma of $\text{DC}\mu\nu$). The following claims hold.

- (1) Suppose $\Gamma \vdash_{\text{DC}\mu\nu} \Delta \mid N : A$ is derivable. Then the following hold.
 - (1a) If $\Gamma, x : A \vdash_{\text{DC}\mu\nu} \Delta \mid M : B$, then $\Gamma \vdash_{\text{DC}\mu\nu} \Delta \mid M[N/x] : B$,
 - (1b) if $K : B \mid \Gamma, x : A \vdash_{\text{DC}\mu\nu} \Delta$, then $K[N/x] : B \mid \Gamma \vdash_{\text{DC}\mu\nu} \Delta$, and
 - (1c) if $\Gamma, x : A \mid S \vdash_{\text{DC}\mu\nu} \Delta$, then $\Gamma \mid S[N/x] \vdash_{\text{DC}\mu\nu} \Delta$.
- (2) Suppose $L : A \mid \Gamma \vdash_{\text{DC}\mu\nu} \Delta$ is derivable. Then the following hold.
 - (2a) If $\Gamma \vdash_{\text{DC}\mu\nu} \Delta, \alpha : A \mid M : B$, then $\Gamma \vdash_{\text{DC}\mu\nu} \Delta \mid M[L/\alpha] : B$,
 - (2b) if $K : B \mid \Gamma \vdash_{\text{DC}\mu\nu} \Delta, \alpha : A$, then $K[L/\alpha] : B \mid \Gamma \vdash_{\text{DC}\mu\nu} \Delta$, and
 - (2c) if $\Gamma \mid S \vdash_{\text{DC}\mu\nu} \Delta, \alpha : A$, then $\Gamma \mid S[L/\alpha] \vdash_{\text{DC}\mu\nu} \Delta$.

Proof. The claims (1a), (1b), and (1c) are shown simultaneously by induction on M , K , and S . The claims (2a), (2b), and (2c) are also shown simultaneously by induction on M , K , and S . □

The duality transformation can be extended from DC to $\text{DC}\mu\nu$.

Definition 3.7 (Duality Transformation). The duality transformation for types, terms, coterms, statements, and inference rule names of $\text{DC}\mu\nu$ is defined by those of DC and the following equations:

$$\begin{aligned} (\mu X.A)^\circ &= \nu X.(A)^\circ, & (\nu X.A)^\circ &= \mu X.(A)^\circ. \\ (\text{in}^{\mu X.A}\langle M \rangle)^\circ &= \text{out}^{\nu X.(A)^\circ}[(M)^\circ], \\ (\text{out}^{\nu X.A}[K])^\circ &= \text{in}^{\mu X.(A)^\circ}\langle (K)^\circ \rangle, \\ (\text{itr}_\alpha^A[K, L])^\circ &= \text{coitr}_{\alpha'}^{(A)^\circ}\langle (K)^\circ, (L)^\circ \rangle, \\ (\text{coitr}_x^A\langle M, N \rangle)^\circ &= \text{itr}_{x'}^{(A)^\circ}[(M)^\circ, (N)^\circ]. \\ (\mu R)^\circ &= (\nu L), & (\nu L)^\circ &= (\mu R), & (\mu L)^\circ &= (\nu R), & (\nu R)^\circ &= (\mu L). \end{aligned}$$

The above duality transformation is well-defined.

Lemma 3.8. *The type $(A)^\circ$ is defined, and $\text{Pos}(A) = \text{Pos}((A)^\circ)$ and $\text{Neg}(A) = \text{Neg}((A)^\circ)$ hold.*

Proof. These claims are shown by induction on A . We consider the cases of $\mu X.B$ and $\nu X.B$. The other cases are straightforwardly proved by the induction hypothesis.

The case of $\mu X.B$: Suppose that $\mu X.B$ is defined. Then we have X is in $\text{Pos}(B)$. By the induction hypothesis, $(B)^\circ$ is defined and X occurs positively in $(B)^\circ$. Therefore $\nu X.(B)^\circ$ is defined, and we have $\text{Pos}(\mu X.B) = \text{Pos}(\nu X.(B)^\circ)$ and $\text{Neg}(\mu X.B) = \text{Neg}(\nu X.(B)^\circ)$ by the induction hypothesis.

The case of $\nu X.B$ can be shown in the similar way to the case of $\mu X.B$. \square

This duality transformation alternates free variables and free covariables that occur in terms and coterms. Let \mathcal{V} be a set of variables, and \mathcal{C} be a set of covariables. Then a set of covariables $(\mathcal{V})^\circ$ is defined by $\{x' \mid x \in \mathcal{V}\}$. A set of variables $(\mathcal{C})^\circ$ is also defined by $\{\alpha' \mid \alpha \in \mathcal{C}\}$.

Lemma 3.9. *Let D be an expression of $\text{DC}\mu\nu$. Then $\text{FV}((D)^\circ) = (\text{FCV}(D))^\circ$ and $\text{FCV}((D)^\circ) = (\text{FV}(D))^\circ$ hold.*

Proof. The claims are shown by induction on D . \square

This duality transformation preserves substitution of types, terms, and coterms.

Lemma 3.10. *Let A and B be types, D be an expression, M be a term, and K be a coterm of $\text{DC}\mu\nu$. Then the following hold.*

- (1) $(A[B/X])^\circ = (A)^\circ[(B)^\circ/X]$.
- (2) $(D[M/x])^\circ = (D)^\circ[(M)^\circ/x']$.
- (3) $(D[K/\alpha])^\circ = (D)^\circ[(K)^\circ/\alpha']$.

Proof. The claim (1) is shown by induction on A . The claims (2) and (3) are shown by induction on D . \square

The extended duality transformation preserves typing, and is an involution in $\text{DC}\mu\nu$.

Proposition 3.11. *The following claims hold.*

- (1) *If J is derived from J_1, \dots, J_n ($n = 1$ or 2) by an inference rule R , then $(J)^\circ$ is derived from $(J_n)^\circ, \dots, (J_1)^\circ$ by the inference rule $(R)^\circ$.*
- (2) *$((A)^\circ)^\circ = A$, $((D)^\circ)^\circ = D$, and $((J)^\circ)^\circ = J$ hold for any type A , expression D , and judgment J of $\text{DC}\mu\nu$.*

Proof. The claim (1) is shown by case analysis of the inference rules of $\text{DC}\mu\nu$ using Lemma 3.10 (1). The claim (2) is shown by induction on types and expressions. \square

Our reduction rules for inductive and coinductive types will be defined so that they correspond to cut elimination procedures in the classical sequent calculus LK extended with inductive definitions and coinductive definitions. In the following proof figures, we will write μ , ν , and $A[B]$ for $\mu X.A$, $\nu X.A$, and $A[B/X]$ respectively. In the logical system, when the cut formula is an inductive type, the cut elimination procedure reduces the proof

$$\frac{\frac{\Gamma \vdash \Delta, A[\mu]}{\Gamma \vdash \Delta, \mu} (\mu R) \quad \frac{\frac{A[B], \Gamma \vdash \Delta, B}{\mu, \Gamma \vdash \Delta} (\mu L) \quad B, \Gamma \vdash \Delta}{\mu, \Gamma \vdash \Delta} (\text{Cut})}{\Gamma \vdash \Delta} (\text{Cut})$$

to the proof

$$\frac{\frac{\frac{\vdots}{\Gamma \vdash \Delta, A[\mu]} (Wk) \quad \frac{\frac{\frac{\vdots}{A[B], \Gamma \vdash \Delta, B} \quad \frac{\vdots}{B, \Gamma \vdash \Delta, B}}{\mu, \Gamma \vdash \Delta, B} (\mu L) \quad \frac{\vdots}{A[B], \Gamma \vdash \Delta, B} (mono)}{\frac{\vdots}{A[\mu], \Gamma \vdash \Delta, B} (Cut)} \quad \frac{\vdots}{B, \Gamma \vdash \Delta} (Cut)}{\Gamma \vdash \Delta, B} (Cut)$$

We can intuitively understand the rule *(mono)* as follows: $\mu \vdash B$ implies $A[\mu] \vdash A[B]$, so we have $A[\mu] \vdash B$ by combining it with $A[B] \vdash B$. This rule will be formally shown in Lemma 3.14 (2a). This reduction changes the cut formula from μ to $A[\mu]$. We do not have to count the cut formula B , since that cut is auxiliary. When the cut formula is a coinductive type, the cut elimination procedure reduces a proof in a dual way to the above reduction.

When we have a function $\lambda x.M$ from A to B and the variable X is in $\text{Pos}(C)$, we can define the function from $C[A/X]$ to $C[B/X]$ by extending $\lambda x.M$. We will use $\text{mono}_{A,B,x.M}^{X,C}\{N\}$ so that this function maps z to $\text{mono}_{A,B,x.M}^{X,C}\{z\}$. We will define $\text{mono}_{A,B,x.M}^{X,C}\{N\}$ by induction on the measure $\|C\|_X$ for a type C and a type variable X , which is defined by induction on C as follows: If X is not free in A , then $\|A\|_X = 0$. In the other cases, we assume that some X occurs in A and we define

$$\begin{aligned}
\|X\|_X &= 1, \\
\|A \wedge B\|_X &= \|A \vee B\|_X = \|A\|_X + \|B\|_X + 1, \\
\|\neg A\|_X &= \|A\|_X + 1, \\
\|\mu Y.A\|_X &= \|\nu Y.A\|_X = \|A\|_X + \|A\|_Y + 1.
\end{aligned}$$

Note that if X is not free in B and we have $X \neq Y$, then $\|A\|_X = \|A[B/Y]\|_X$.

The number $\|A\|_X$ will also be used for evaluating the size of $\text{mono}_{A,B,x.M}^{X,C}\{N\}$ by using M , N , and C (see Lemma 6.2). If we replaced $\|A\|_X + \|A\|_Y + 1$ by $\|A\|_X + 1$ in the definition of $\|\mu Y.A\|_X$ and $\|\nu Y.A\|_X$, it would not work for this purpose.

Definition 3.12. Assume a type variable X and types A, B, C are given and X is not free in A and B . For a variable x and terms M and N , we define the term $\text{mono}_{A,B,x.M}^{X,C}\{N\}$ by induction on $\|C\|_X$ as follows:

$$\begin{aligned}
\text{mono}_{A,B,x.M}^{X,X}\{N\} &= (N \bullet x.(M \bullet \alpha)).\alpha, \\
\text{mono}_{A,B,x.M}^{X,C}\{N\} &= N \quad (X \text{ does not occur in } C), \\
\text{mono}_{A,B,x.M}^{X,C \wedge D}\{N\} &= \langle \text{mono}_{A,B,x.M}^{X,C}\{(N \bullet \text{fst}[\alpha]).\alpha\}, \text{mono}_{A,B,x.M}^{X,D}\{(N \bullet \text{snd}[\beta]).\beta\} \rangle, \\
\text{mono}_{A,B,x.M}^{X,C \vee D}\{N\} &= (N \bullet [y.(\langle \text{mono}_{A,B,x.M}^{X,C}\{y\} \rangle \text{inl} \bullet \gamma), z.(\langle \text{mono}_{A,B,x.M}^{X,D}\{z\} \rangle \text{inr} \bullet \gamma)]).\gamma, \\
\text{mono}_{A,B,x.M}^{X,\neg C}\{N\} &= [z.(N \bullet \text{not}(\langle \text{mono}_{A,B,x.M}^{X,C}\{z\} \rangle))]\text{not}, \\
\text{mono}_{A,B,x.M}^{X,\mu Y.C}\{N\} &= (N \bullet \text{itr}_\alpha^{\mu Y.C[B/X]}[z.(\text{in}^{\mu Y.C[B/X]}(\langle \text{mono}_{A,B,x.M}^{X,C[\mu Y.C[B/X]/Y]}\{z\} \rangle \bullet \alpha), \beta)].\beta, \\
\text{mono}_{A,B,x.M}^{X,\nu Y.C}\{N\} &= \text{coitr}_z^{\nu Y.C[A/X]}(\langle \text{mono}_{A,B,x.M}^{X,C[\nu Y.C[A/X]/Y]}\{(z \bullet \text{out}^{\nu Y.C[A/X]}[\alpha]).\alpha\}, N \rangle.
\end{aligned}$$

For a covariable α and coterms K and L , we also define

$$\text{mono}_{A,B,\alpha,K}^{X,C}\{L\} = (\text{mono}_{(B)^\circ, (A)^\circ, \alpha', (K)^\circ}^{X,(C)^\circ}\{(L)^\circ\})^\circ.$$

Note that $\|\mu Y.C\|_X > \|C[\mu Y.C[B/X]/Y]\|_X$ and $\|\nu Y.C\|_X > \|C[\nu Y.C[A/X]/Y]\|_X$ hold since X is not free in $\mu Y.C[B/X]$ and $\nu Y.C[A/X]$. We cannot replace $C[\mu Y.C[B/X]/Y]$ by C in the definition of $\text{mono}_{A,B,x.M}^{X,\mu Y.C}\{N\}$ because of the type annotation for in . For readability, we sometimes write $\text{mono}_{A,B,x}^{X,C}\{M, N\}$ and $\text{mono}_{A,B,\alpha}^{X,C}\{K, L\}$ for $\text{mono}_{A,B,x.M}^{X,C}\{N\}$ and $\text{mono}_{A,B,\alpha,K}^{X,C}\{L\}$, respectively.

The paper [17] studied an intuitionistic logical system with strictly-positive inductive definitions, and on the other hand we study a classical logical system with positive inductive definitions. Our cut elimination for inductive types is the same as theirs, and on the other hand our cut elimination for coinductive types is different from theirs. They can avoid the use of `mono`. However, we cannot straightforwardly compare our method and their method, since our system is strictly larger than their system.

Our method works only for classical logic and does not work for an intuitionistic logic. This is because our cut elimination procedure keeps the duality and we have the corresponding proof rule that manipulates a succedent if we have some proof rule that manipulates an antecedent. In particular, we define the operator `mono` for coterms as the dual of the operator `mono` for terms. Roughly speaking, in the proof of the next lemma, when we show the properties of `mono` for negation by using the derivation

$$\frac{\frac{A \vdash B}{A, \neg B \vdash}}{\neg B \vdash \neg A},$$

we need the following derivation in order to show the properties of its dual:

$$\frac{\frac{B \vdash A}{\vdash A, \neg B}}{\neg A \vdash \neg B}$$

which uses a non-intuitionistic sequent.

Lemma 3.13. *The following claims hold.*

- (1a) $FV(\text{mono}_{A,B,x}^{X,C}\{M, N\}) \subseteq (FV(M) \setminus \{x\}) \cup FV(N)$.
- (1b) $FCV(\text{mono}_{A,B,x}^{X,C}\{M, N\}) \subseteq FCV(M) \cup FCV(N)$.
- (2a) $FV(\text{mono}_{A,B,\alpha}^{X,C}\{K, L\}) \subseteq FV(K) \cup FV(L)$.
- (2b) $FCV(\text{mono}_{A,B,\alpha}^{X,C}\{K, L\}) \subseteq (FCV(K) \setminus \{\alpha\}) \cup FCV(L)$.

Proof. The claims (1a) and (1b) are shown by induction on $\|C\|_X$. The claims (2a) and (2b) are shown by using (1a), (2b), and Lemma 3.9. \square

Lemma 3.14. *Assume X is in $\text{Pos}(C)$ and $\text{Neg}(D)$. Then the following hold:*

- (1a) $\Gamma, x: A \vdash \Delta \mid M: B \text{ and } \Gamma \vdash \Delta \mid N: C[A] \text{ implies } \Gamma \vdash \Delta \mid \text{mono}_{A,B,x,M}^{X,C}\{N\}: C[B],$
- (1b) $\Gamma, x: B \vdash \Delta \mid M: A \text{ and } \Gamma \vdash \Delta \mid N: D[A] \text{ implies } \Gamma \vdash \Delta \mid \text{mono}_{A,B,x,M}^{X,D}\{N\}: D[B],$
- (2a) $K: A \mid \Gamma \vdash \Delta, \alpha: B \text{ and } L: C[B] \mid \Gamma \vdash \Delta \text{ implies } \text{mono}_{A,B,\alpha,K}^{X,C}\{L\}: C[A] \mid \Gamma \vdash \Delta,$
- (2b) $K: B \mid \Gamma \vdash \Delta, \alpha: A \text{ and } L: D[B] \mid \Gamma \vdash \Delta, \text{ implies } \text{mono}_{A,B,\alpha,K}^{X,D}\{L\}: C[A] \mid \Gamma \vdash \Delta,$

where $C[A]$ and $D[A]$ are abbreviations of $C[A/X]$ and $D[A/X]$, respectively.

Proof. The claims (1a) and (1b) are shown simultaneously by induction on $\|C\|_X$ and $\|D\|_X$. The claims (2a) and (2b) are shown by using (1a), (1b), and Proposition 3.11. \square

The following proposition is obtained as a special case of the above lemma.

Proposition 3.15. *Assume X is in $\text{Pos}(C)$. The following are derivable:*

$$\frac{\Gamma, x : A \vdash \Delta \mid M : B}{\Gamma, z : C[A] \vdash \Delta \mid \text{mono}_{A,B,x,M}^{X,C}\{z\} : C[B]} \quad \frac{K : A \mid \Gamma \vdash \Delta, \alpha : B}{\text{mono}_{A,B,\alpha,K}^{X,C}\{\beta\} : C[A] \mid \Gamma \vdash \Delta, \beta : C[B]}$$

Definition 3.16. We define the one-step reduction relation $\longrightarrow_{\text{DC}\mu\nu}$ of $\text{DC}\mu\nu$ as the compatible closure of the reduction rules of DC and the following reduction rules:

$$\begin{aligned} (\beta\mu) \quad & \text{in}^{\mu X.C}\langle M \rangle \bullet \text{itr}_\alpha^A[K, L] \longrightarrow_{\text{DC}\mu\nu} (M \bullet \text{mono}_{\mu X.C, A, \beta}^{X,C}\{\text{itr}_\alpha^A[K, \beta], K\}).\alpha \bullet L, \\ (\beta\nu) \quad & \text{coitr}_x^A\langle M, N \rangle \bullet \text{out}^{\nu X.C}[K] \longrightarrow_{\text{DC}\mu\nu} N \bullet x.(\text{mono}_{A, \nu X.C, z}^{X,C}\{\text{coitr}_x^A\langle M, z \rangle, M\} \bullet K). \end{aligned}$$

This system has subject reduction.

Proposition 3.17 (Subject reduction of $\text{DC}\mu\nu$). *The following claims hold.*

- (1) *If $\Gamma \vdash_{\text{DC}\mu\nu} \Delta \mid M : A$ and $M \longrightarrow_{\text{DC}\mu\nu} N$, then $\Gamma \vdash_{\text{DC}\mu\nu} \Delta \mid N : A$ holds.*
- (2) *If $K : A \mid \Gamma \vdash_{\text{DC}\mu\nu} \Delta$ and $K \longrightarrow_{\text{DC}\mu\nu} L$, then $L : A \mid \Gamma \vdash_{\text{DC}\mu\nu} \Delta$ holds.*
- (3) *If $\Gamma \mid S \vdash_{\text{DC}\mu\nu} \Delta$ and $S \longrightarrow_{\text{DC}\mu\nu} T$, then $\Gamma \mid T \vdash_{\text{DC}\mu\nu} \Delta$ holds.*

Proof. They are shown simultaneously by induction on the generation of $\longrightarrow_{\text{DC}\mu\nu}$ using Lemma 3.4, 3.5, 3.6, and 3.14. We consider the cases of $(\beta\mu)$ and $(\beta\nu)$.

Case of $(\beta\mu)$. Assume $\Gamma \mid \text{in}^{\mu X.C}\langle M \rangle \bullet \text{itr}_\alpha^A[K, L] \vdash \Delta$ is derivable in $\text{DC}\mu\nu$. We use μ and $C[A]$ as abbreviations of $\mu X.C$ and $C[A/X]$, respectively. The last rule of the derivation must be (Cut) rule. Then $\Gamma \vdash \Delta \mid \text{in}^{\mu X.C}\langle M \rangle : D$ and $\text{itr}_\alpha^A[K, L] : D \mid \Gamma \vdash \Delta$ are derivable for some type D . Since the last rules of these derivations must be (μR) and (μL) , we obtain D is $\mu X.C$, and the derivations of $\Gamma \vdash \Delta \mid M : C[\mu]$, the judgment $K : C[A] \mid \Gamma \vdash \Delta, \alpha : B$, and $L : A \mid \Gamma \vdash \Delta, \alpha : B$. Hence we have $\text{itr}_\alpha^A[K, \beta] : \mu \mid \Gamma \vdash \Delta, \beta : A$ by (AxL) and (μL) rules, and then $\text{mono}_{\mu, A, \beta}^{X,C}\{\text{itr}_\alpha^A[K, \beta], K\} : C[\mu] \mid \Gamma \vdash \Delta, \alpha : A$ is derivable by Lemma 3.14. Therefore we have $\Gamma \mid (M \bullet \text{mono}_{\mu, A, \beta}^{X,C}\{\text{itr}_\alpha^A[K, \beta], K\}).\alpha \bullet L \vdash \Delta$ by using (IR) , (Cut) rules.

The case of $(\beta\nu)$ is shown similarly to the case of $(\beta\mu)$.

The other cases are straightforwardly proved by the induction hypothesis. \square

The duality transformation $(-)^{\circ}$ preserves reduction.

Theorem 3.18 (Duality of $\text{DC}\mu\nu$). *$D \longrightarrow_{\text{DC}\mu\nu} E$ implies $(D)^{\circ} \longrightarrow_{\text{DC}\mu\nu} (E)^{\circ}$ for any expressions D and E .*

Proof. This is proved by induction on the generation of $\longrightarrow_{\text{DC}\mu\nu}$. \square

Proposition 3.19. *If $D \longrightarrow_{\text{DC}\mu\nu} E$ is the rules $(\beta\mu)$ and $(\beta\nu)$, then $(D)^{\circ} \longrightarrow_{\text{DC}\mu\nu} (E)^{\circ}$ is $(\beta\nu)$ and $(\beta\mu)$ respectively.*

We have shown the duality of inductive types and coinductive types. Proposition 3.11 and Theorem 3.18 show that the duality transformation is a homomorphic involution. The description of a type can be defined as the set of the type itself, its terms, its coterm, and their reduction. The duality transformation maps the description of an inductive type and that of a coinductive type to each other. That is, we have the following. (1) Definition 3.7 shows that the inductive type $\mu X.A$ is mapped to the coinductive type $\nu X.(A)^{\circ}$, the term constructed by in for the inductive type is mapped to the coterm constructed by out for the coinductive type, and the coterm constructed by itr for the inductive type is mapped to the term constructed by coitr for the coinductive type. (2) Proposition 3.19 shows that the cut elimination procedure of the inductive type is mapped to the cut elimination procedure of the coinductive type. (3) the coinductive type is mapped to the inductive type in a similar way to (1) and (2).

Remark 3.20. We cannot define our typing system by using

$$\frac{K : C[A/X] \mid \Gamma \vdash \Delta, \alpha : A}{\text{itr}_\alpha^A[K, \beta] : \mu X. C \mid \Gamma \vdash \Delta, \beta : A} (\mu L')$$

instead of the typing rule (μL) . If we used $(\mu L')$, the set of terms would not be closed under substitution, because $\text{itr}_\alpha^A[K, L]$ would not have typing rules for it and hence it would not be a term, though it is obtained from $\text{itr}_\alpha^A[K, \beta]$ by substituting L for β .

4. EXAMPLES

In this section we show some examples of inductive and coinductive types in $\text{DC}\mu\nu$. Let X_0 be a distinguished type variable. We use the following abbreviations:

$$\top = \neg X_0 \vee X_0, \quad \perp = \neg X_0 \wedge X_0, \quad \text{and} \quad * = \lambda x. x.$$

The type Nat of natural numbers can be represented by:

$$\begin{aligned} \text{Nat} &= \mu X. (\top \vee X), \\ \mathbf{0} &= \text{in}^{\text{Nat}} \langle \langle * \rangle \text{inl} \rangle, \\ \text{succ} \langle M \rangle &= \text{in}^{\text{Nat}} \langle \langle M \rangle \text{inr} \rangle, \end{aligned}$$

where $\mathbf{0}$ is the zero and succ is the successor. We can prove $\Gamma \vdash \Delta \mid \mathbf{0} : \text{Nat}$. We can also prove $\Gamma \vdash \Delta \mid \text{succ} \langle M \rangle : \text{Nat}$ from $\Gamma \vdash \Delta \mid M : \text{Nat}$. The n -th natural number \tilde{n} is represented by $\text{succ} \langle \text{succ} \langle \dots \text{succ} \langle \mathbf{0} \rangle \dots \rangle \rangle$ (n times of succ). We will write $M[-/x]^n(N)$ for $M[M[\dots [M[N/x]/x] \dots /x]/x]$ (n times of M). We define a coterm $\text{Itr}^B[F, N, K]$ of type Nat by $\text{itr}_\alpha^B[y.(N \bullet \alpha), x.(F \bullet (x @ \alpha))], K]$, where y is not free in N , the term F has type $B \supset B$, and N and K are of type B . When the coterm $\text{Itr}^B[F, N, K]$ gets \tilde{n} as its input, it computes n -time iterations of applying the function F to N , and passes the output to K . This reduces $\tilde{n} \bullet \text{Itr}^B[\lambda x. M, N, K]$ to $M[-/x]^n(N) \bullet K$.

The type $\text{List}(A)$ of lists of elements of type A is represented by:

$$\begin{aligned} \text{List}(A) &= \mu X. (\top \vee (A \wedge X)), \\ \text{nil} &= \text{in}^{\text{List}(A)} \langle \langle * \rangle \text{inl} \rangle, \\ M :: Nl &= \text{in}^{\text{List}(A)} \langle \langle \langle M, Nl \rangle \text{inr} \rangle \rangle. \end{aligned}$$

The term nil is the empty list and $::$ is the list constructor. In $\text{DC}\mu\nu$, the judgment $\Gamma \vdash \Delta \mid \text{nil} : \text{List}(A)$ is provable. The judgment $\Gamma \vdash \Delta \mid M :: Nl : \text{List}(A)$ is also provable from $\Gamma \vdash \Delta \mid M : A$ and $\Gamma \vdash \Delta \mid Nl : \text{List}(A)$.

We note that the above examples can be considered under the call-by-value setting (section 7) if we restrict terms in the above examples to values.

We can also define the type $\text{Stream}(A)$ of streams of elements of type A by:

$$\begin{aligned} \text{Stream}(A) &= \nu X. (A \wedge X), \\ \text{cons} \langle M, Ns \rangle &= \text{coitr}_x^{A \wedge \text{Stream}(A)} \langle \langle \pi_1(x), (\pi_2(x) \bullet \text{out}^{\text{Stream}(A)}[\alpha]).\alpha \rangle, \langle M, Ns \rangle \rangle, \\ \text{hd}[K] &= \text{out}^{\text{Stream}(A)}[\text{fst}[K]], \\ \text{tl}[L] &= \text{out}^{\text{Stream}(A)}[\text{snd}[L]], \end{aligned}$$

where $\pi_1(M)$ is the first projection of M defined by $(M \bullet \text{fst}[\alpha]).\alpha$, and $\pi_2(M)$ is the second projection of M defined by $(M \bullet \text{snd}[\alpha]).\alpha$. The term $\text{cons} \langle M, Ns \rangle$ constructs a new stream from a

given element M and a given stream Ns . The coterm $\text{hd}[K]$ receives the first element from a given stream and gives it to K . The coterm $\text{tl}[L]$ removes the first element from a given stream and gives the resulting stream to L . We can prove $\Gamma \vdash \Delta \mid \text{cons}\langle M, Ns \rangle : \text{Stream}(A)$ from $\Gamma \vdash \Delta \mid M : A$ and $\Gamma \vdash \Delta \mid Ns : \text{Stream}(A)$. We can also prove $\text{hd}[K] : \text{Stream}(A) \mid \Gamma \vdash \Delta$ from $K : A \mid \Gamma \vdash \Delta$. We can also prove $\text{tl}[L] : \text{Stream}(A) \mid \Gamma \vdash \Delta$ from $L : \text{Stream}(A) \mid \Gamma \vdash \Delta$. This reduces $\text{cons}\langle M, Ns \rangle \bullet \text{hd}[K]$ to $M \bullet K$. We also reduce $\text{cons}\langle M, Ns \rangle \bullet \text{tl}^{n+1}[\text{hd}[K]]$ to $Ns \bullet \text{tl}^n[\text{hd}[K]]$, where $\text{tl}^n[\text{hd}[K]]$ is defined by $\text{tl}[\text{tl}[\dots \text{tl}[\text{hd}[K]] \dots]]$ (n times of tl). Hence the coterm $\text{tl}^n[\text{hd}[K]]$ receives the n -th element of a given stream and gives it to K . Let M be a term of type A . The stream of infinite number of M is represented by $\text{coitr}_x^\top \langle \langle M, x \rangle, * \rangle$, where x is a fresh variable. We will write $\text{stream}(M)$ for $\text{coitr}_x^\top \langle \langle M, x \rangle, * \rangle$. Indeed, the statement $\text{stream}(M) \bullet \text{tl}^n[\text{hd}[K]]$ is reduced to M for any n . This means that any n -th element of $\text{stream}(M)$ is M .

We note that this stream example can be considered under the call-by-name setting (section 7) if we restrict coterminals in the above example to covalues.

Proposition 4.1. *Nat is dual to $\text{Stream}(\perp)$, that is, $(\text{Nat})^\circ = \text{Stream}(\perp)$, $(\emptyset)^\circ = \text{hd}[(*)^\circ]$, and $(\text{succ}\langle M \rangle)^\circ = \text{tl}[(M)^\circ]$ hold.*

If $\text{Stream}(\perp)$ is considered under the call-by-name setting and Nat is considered under the call-by-value setting, then the duality of the above proposition can be understood as follows. The type \top means the singleton set $\{*\}$. The type \perp means the type of a program that returns some answer after computation with the input $*$ since \perp is equivalent to $\neg\top$. The type Nat means the infinite disjoint sum $\top + \top + \top + \dots$. The type $\text{Stream}(\perp)$ means the infinite cartesian product $\perp \times \perp \times \perp \times \dots$. Since a term in $\text{Stream}(\perp)$ is equivalent to a coterm in Nat , when the term gets some natural number and is computed, it returns some answer. When the term gets the natural number \tilde{n} , since \tilde{n} is $*$ in the n -th \top in $\top + \top + \dots$, the term in the n -th \perp in $\perp \times \perp \times \dots$ is given the input $*$ and it is computed to give some answer.

Here we can also consider examples that include non-deterministic choices. Let M and N be terms of same type, x be a fresh variable, α and β be fresh covariables. We define the non-deterministic choice $\langle M \mid N \rangle$ by

$$\langle M \mid N \rangle = ((M \bullet \alpha) \bullet \beta \bullet x.(N \bullet \alpha)).\alpha,$$

where α and β are fresh covariables. This term has both (β_L) and (β_R) -redexes. It is reduced to M if the (β_R) -redex is chosen, and is reduced to N if the (β_L) -redex is chosen. Thus, $\langle M \mid N \rangle$ can be considered as a non-deterministic choice of either M or N . This non-deterministic choice $\langle M \mid N \rangle$ is forced to choose M under the call-by-value strategy, and is forced to choose N under the call-by-name strategy.

An example of non-deterministic computation is the list insertion function. This function gets a list as its input data, and non-deterministically chooses one arbitrary place in the list. Then it returns a new list that is obtained by inserting a given element at the place.

Let M be a term of type A , and K' be a coterm of type $\text{List}(A) \wedge \text{List}(A)$. Then we define $\text{ins}_M[K']$ of type $\text{List}(A)$ by

$$\begin{aligned} \text{ins}_M[K'] &= \text{itr}_\alpha^{\text{List}(A) \wedge \text{List}(A)}[[L_1(\alpha), L_2(\alpha)], K'], \\ L_1(\alpha) &= x.(\langle M :: \text{nil}, \text{nil} \rangle \bullet \alpha), \\ L_2(\alpha) &= z.(\langle \langle \pi_1(z) :: \pi_1\pi_2(z), \pi_1(z) :: \pi_2\pi_2(z) \rangle \rangle \mid \langle M :: \pi_1(z) :: \pi_2\pi_2(z), \pi_1(z) :: \pi_1\pi_2(z) \rangle \rangle \bullet \alpha) \end{aligned}$$

where x occurs in $L_1(\alpha)$ is a fresh variable of type \top , and z occurs in $L_2(\alpha)$ is a fresh variable of type $A \wedge (\text{List}(A) \wedge \text{List}(A))$. Then if Nl is a list and Nl' is a list obtained by inserting M in some place of Nl , then the statement $Nl \bullet \text{ins}_M[K']$ can be reduced to $\langle Nl', Nl \rangle \bullet K'$. We can show this by induction on the length of Nl . If Nl is nil , then Nl' is $M :: \text{nil}$. The statement $\text{nil} \bullet \text{ins}_M[K']$

is reduced to $\langle M :: \text{nil}, \text{nil} \rangle \bullet K'$. If Nl is $N :: Nl_0$, then Nl' is either $N :: Nl'_0$ or $M :: N :: Nl_0$, where Nl'_0 is an inserted list obtained from Nl_0 . The statement $(N :: Nl_0) \bullet \text{ins}_M[K']$ is reduced to $\langle \langle N, (Nl_0 \bullet \text{ins}_M[\gamma]).\gamma \rangle \bullet L_2(\alpha) \rangle.\alpha \bullet K'$. Then this statement is reduced to $\langle \langle N, \langle Nl_0, Nl'_0 \rangle \rangle \bullet L_2(\alpha) \rangle.\alpha \bullet K'$ by the induction hypothesis. We have $\langle \langle N :: Nl'_0, Nl \rangle \mid \langle M :: N :: Nl_0, Nl \rangle \rangle \bullet K'$. Hence we can obtain $\langle M :: N :: Nl_0, Nl \rangle \bullet K'$ or $\langle M :: N :: Nl_0, Nl \rangle \bullet K'$.

Let K be a coterms of type $\text{List}(A)$. Here we define

$$\text{insert}_M[K] = \text{ins}_M[\text{fst}[K]].$$

Then the statement $Nl \bullet \text{insert}_M[K]$ is reduced to $Nl' \bullet K$ for any inserted list Nl' obtained from Nl .

5. THE SECOND-ORDER DUAL CALCULUS DC2

We consider the second-order extension DC2 of DC given by Tzevelekos [26]. He showed the basic properties of DC2, such as the substitution lemma and subject reduction. Without formal discussion, he also mentioned that his translation from DC into the symmetric λ -calculus can be extended to the second-order case. In this section, we give a formal definition of the second-order translation from DC2 into the second-order symmetric λ -calculus, and show the strong normalization of DC2 by using this translation. For this purpose we will use the strong normalization result of the second-order symmetric λ -calculus given in [21].

Definition 5.1. An expression is defined to be strongly normalizing if there does not exist any infinite reduction sequence starting from the expression.

First, we define a second-order extension DC2 of DC.

Definition 5.2 (DC2). The types, terms, coterms, and statements of DC2 are defined by:

Types	$A ::= X \mid A \wedge A \mid A \vee A \mid \neg A \mid \forall X.A \mid \exists X.A,$
Terms	$M ::= x \mid \langle M, M \rangle \mid \langle M \rangle \text{inl} \mid \langle M \rangle \text{inr} \mid [K] \text{not} \mid (S).\alpha \mid \langle M \rangle \mathbf{a} \mid \langle M \rangle \mathbf{e},$
Coterms	$K ::= \alpha \mid [K, K] \mid \text{fst}[K] \mid \text{snd}[K] \mid \text{not}\langle M \rangle \mid x.(S) \mid \mathbf{a}[K] \mid \mathbf{e}[K],$
Statements	$S ::= M \bullet K.$

The typing rules and reduction rules (denoted by $\longrightarrow_{\text{DC2}}$) of DC2 are defined by extending the rules of DC with the following rules:

$$\begin{array}{c}
\frac{\Gamma \vdash \Delta \mid M : A}{\Gamma \vdash \Delta \mid \langle M \rangle \mathbf{a} : \forall Z.A} (\forall R) \quad \frac{K : A[B/X] \mid \Gamma \vdash \Delta}{\mathbf{a}[K] : \forall X.A \mid \Gamma \vdash \Delta} (\forall L) \\
\frac{\Gamma \vdash \Delta \mid M : A[B/X]}{\Gamma \vdash \Delta \mid \langle M \rangle \mathbf{e} : \exists X.A} (\exists R) \quad \frac{K : A \mid \Gamma \vdash \Delta}{\mathbf{e}[K] : \exists Z.A \mid \Gamma \vdash \Delta} (\exists L) \\
(\beta\forall) \quad \langle M \rangle \mathbf{a} \bullet \mathbf{a}[K] \longrightarrow_{\text{DC2}} M \bullet K, \\
(\beta\exists) \quad \langle M \rangle \mathbf{e} \bullet \mathbf{e}[K] \longrightarrow_{\text{DC2}} M \bullet K,
\end{array}$$

where Z is not free in Γ and Δ in $(\forall R)$ and $(\exists L)$. We write $\longrightarrow_{\text{DC2}}^+$ to denote the transitive closure of $\longrightarrow_{\text{DC2}}$.

We have the new constructors \mathbf{a} and \mathbf{e} , which are trivial witnesses for the quantifiers at the level of expressions, so that the system has subject reduction. We choose our DC2 so that it does not contain type information in expressions, since our purpose is to show strong normalization of the second-order dual calculus, and in general the strong normalization of the system with type information is implied by the strong normalization of the system without type information.

We sometimes use the symbol \vdash_{DC2} instead of the symbol \vdash that appears in a judgment in order to explicitly show it is a judgment of DC2. That is, we write $\Gamma \vdash_{\text{DC2}} \Delta \mid M : A$ for the judgment $\Gamma \vdash \Delta \mid M : A$. Similarly, we write $K : A \mid \Gamma \vdash_{\text{DC2}} \Delta$ and $\Gamma \mid S \vdash_{\text{DC2}} \Delta$.

We write $\Gamma[B/X]$ for $x_1 : C_1[B/X], \dots, x_n : C_n[B/X]$ if Γ is $x_1 : C_1, \dots, x_n : C_n$. We also write $\Delta[B/X]$ for $\alpha_1 : D_1[B/X], \dots, \alpha_m : D_m[B/X]$ if Δ is $\alpha_1 : D_1, \dots, \alpha_m : D_m$.

Lemma 5.3. *The following claims hold.*

- (1) *If $\Gamma \vdash_{\text{DC2}} \Delta \mid M : A$, then $\Gamma[B/X] \vdash_{\text{DC2}} \Delta[B/X] \mid M : A[B/X]$ holds.*
- (2) *If $K : A \mid \Gamma \vdash_{\text{DC2}} \Delta$, then $K : A \mid \Gamma[B/X] \vdash_{\text{DC2}} \Delta[B/X]$ holds.*
- (3) *If $\Gamma \mid S \vdash_{\text{DC2}} \Delta$, then $\Gamma[B/X] \mid S \vdash_{\text{DC2}} \Delta[B/X]$ holds.*

Proof. They are shown simultaneously by induction on expressions. \square

The basic lemmas for DC and $\text{DC}\mu\nu$ are also shown in DC2. We use Lemma 5.3 to show weakening lemma.

Lemma 5.4 (Weakening lemma). *Let $\Gamma \subseteq \Gamma'$ and $\Delta \subseteq \Delta'$. Then the following hold in DC2.*

- (1) *If $\Gamma \vdash \Delta \mid M : A$ is provable, then $\Gamma' \vdash \Delta' \mid M : A$ holds.*
- (2) *If $K : A \mid \Gamma \vdash \Delta$ is provable, then $K : A \mid \Gamma' \vdash \Delta'$ holds.*
- (3) *If $\Gamma \mid S \vdash \Delta$ is provable, then $\Gamma' \mid S \vdash \Delta'$ holds.*

Proof. They are shown simultaneously by induction on M , K , and S . We use Lemma 5.3 when we show the cases of $\langle M \rangle a$ and $e[K]$. We consider these cases.

The case of $\langle M \rangle a$. Assume $\Gamma \subseteq \Gamma'$, $\Delta \subseteq \Delta'$, and $\Gamma \vdash \Delta \mid \langle M \rangle a : A$ is derivable. Since the last rule of the derivation must be $(\forall R)$, we have A is $\forall X.B$ for some B , the variable X is not free in Γ and Δ , and $\Gamma \vdash \Delta \mid M : B$ is derivable. Then we have $\Gamma \vdash \Delta \mid M : B[Z/X]$ for a fresh type variable Z by using Lemma 5.3. By the induction hypothesis, $\Gamma' \vdash \Delta' \mid M : B[Z/X]$ holds. Therefore we obtain $\Gamma' \vdash \Delta' \mid \langle M \rangle a : \forall Z.(B[Z/X])$ by $(\forall R)$ rule, since Z is not free in Γ' and Δ' .

The case of $e[K]$ is shown similar to the case of $\langle M \rangle a$.

The other cases are straightforwardly proved by the induction hypothesis. \square

Lemma 5.5. *Let $\Gamma' \subseteq \Gamma$ and $\Delta' \subseteq \Delta$. Then the following hold in DC2.*

- (1) *If $FV(M) \subseteq \text{dom}(\Gamma')$ and $FCV(M) \subseteq \text{dom}(\Delta')$, then $\Gamma \vdash \Delta \mid M : A$ implies $\Gamma' \vdash \Delta' \mid M : A$.*
- (2) *If $FV(K) \subseteq \text{dom}(\Gamma')$ and $FCV(K) \subseteq \text{dom}(\Delta')$, then $K : A \mid \Gamma \vdash \Delta$ implies $K : A \mid \Gamma' \vdash \Delta'$.*
- (3) *If $FV(S) \subseteq \text{dom}(\Gamma')$ and $FCV(S) \subseteq \text{dom}(\Delta')$, then $\Gamma \mid S \vdash \Delta$ implies $\Gamma' \mid S \vdash \Delta'$.*

Proof. They are shown simultaneously by induction on M , K , and S . \square

Lemma 5.6 (Substitution lemma). *The following claims hold.*

- (1) *Suppose $\Gamma \vdash_{\text{DC2}} \Delta \mid N : A$ is derivable. Then the following hold.*
 - (1a) *If $\Gamma, x : A \vdash_{\text{DC2}} \Delta \mid M : B$, then $\Gamma \vdash_{\text{DC2}} \Delta \mid M[N/x] : B$.*
 - (1b) *If $K : B \mid \Gamma, x : A \vdash_{\text{DC2}} \Delta$, then $K[N/x] : B \mid \Gamma \vdash_{\text{DC2}} \Delta$.*
 - (1c) *If $\Gamma, x : A \mid S \vdash_{\text{DC2}} \Delta$, then $\Gamma \mid S[N/x] \vdash_{\text{DC2}} \Delta$.*
- (2) *Suppose $L : A \mid \Gamma \vdash_{\text{DC2}} \Delta$ is derivable. Then the following hold.*
 - (2a) *If $\Gamma \vdash_{\text{DC2}} \Delta, \alpha : A \mid M : B$, then $\Gamma \vdash_{\text{DC2}} \Delta \mid M[L/\alpha] : B$.*
 - (2b) *If $K : B \mid \Gamma \vdash_{\text{DC2}} \Delta, \alpha : A$, then $K[L/\alpha] : B \mid \Gamma \vdash_{\text{DC2}} \Delta$.*
 - (2c) *If $\Gamma \mid S \vdash_{\text{DC2}} \Delta, \alpha : A$, then $\Gamma \mid S[L/\alpha] \vdash_{\text{DC2}} \Delta$.*

Proof. The claims (1a), (1b), and (1c) are shown simultaneously by induction on M , K , and S . The claims (2a), (2b), and (2c) are also shown simultaneously by induction on M , K , and S . \square

This system has subject reduction.

Proposition 5.7 (Subject reduction of DC2). *The following claims hold.*

- (1) *If $\Gamma \vdash_{\text{DC2}} \Delta \mid M : A$ and $M \longrightarrow_{\text{DC2}} N$, then $\Gamma \vdash_{\text{DC2}} \Delta \mid N : A$ holds.*
- (2) *If $K : A \mid \Gamma \vdash_{\text{DC2}} \Delta$ and $K \longrightarrow_{\text{DC2}} L$, then $L : A \mid \Gamma \vdash_{\text{DC2}} \Delta$ holds.*
- (3) *If $\Gamma \mid S \vdash_{\text{DC2}} \Delta$ and $S \longrightarrow_{\text{DC2}} T$, then $\Gamma \mid T \vdash_{\text{DC2}} \Delta$ holds.*

Proof. They are shown simultaneously by induction on the generation of $\longrightarrow_{\text{DC2}}$ using Lemma 5.3, 5.4, 5.5, and 5.6. We show the cases of $(\beta\forall)$ and $(\beta\exists)$.

The case of $(\beta\forall)$. Suppose $\Gamma \mid \langle M \rangle a \bullet a[K] \vdash \Delta$ is derivable. Then $\Gamma \vdash \Delta \mid \langle M \rangle a : C$ and $a[K] : C \mid \Gamma \vdash \Delta$ are derivable for some type C . Since the last rules of these derivation must be $(\forall R)$ and $(\forall L)$, we have C is $\forall X.A$ for some A , X is not free in both Γ and Δ , and $\Gamma \vdash \Delta \mid M : A$ and $K : A[B/X] \mid \Gamma \vdash \Delta$ are derivable for some B . Then we can obtain $\Gamma \vdash \Delta \mid M : A[B/X]$ by Lemma 5.3. Therefore $\Gamma \mid M \bullet K \vdash \Delta$ can be derived by (Cut) rule.

The case of $(\beta\exists)$ is shown similar to the case of $(\beta\forall)$.

The other cases are straightforwardly proved by the induction hypothesis. \square

Remark 5.8. The trivial witnesses a and e are necessary for the subject reduction. If we did not have these constructors, the subject reduction would fail. If we chose the following $(\forall R')$ and $(\forall L')$ instead of $(\forall R)$ and $(\forall L)$,

$$\frac{\Gamma \vdash \Delta \mid M : A}{\Gamma \vdash \Delta \mid M : \forall Z.A} (\forall R') \quad \frac{K : A[B/X] \mid \Gamma \vdash \Delta}{K : \forall X.A \mid \Gamma \vdash \Delta} (\forall L')$$

then the following would be a counter-example: we would have $\Gamma \mid (x \bullet \text{fst}[\alpha]).\alpha \bullet \beta \vdash \Delta$ where Γ is $x : X \wedge Y$, the sequence Δ is $\beta : \forall Z.X$, and $Z \neq X, Y$, but would not have $\Gamma \mid x \bullet \text{fst}[\beta] \vdash \Delta$, though $(x \bullet \text{fst}[\alpha]).\alpha \bullet \beta$ is reduced to $x \bullet \text{fst}[\beta]$.

In λ -calculus the constructor a is not necessary for subject reduction while the constructor e is necessary for it [25]. In our system, since \forall and \exists are dual, the constructor a is also needed.

The duality transformation can be extended from DC to DC2.

Definition 5.9 (Duality Transformation). The duality transformation for types, expressions, and inference rule names of DC2 is defined by those of DC and the following equations:

$$\begin{aligned} (\forall X.A)^\circ &= \exists X.(A)^\circ, & (\exists X.A)^\circ &= \forall X.(A)^\circ, \\ (\langle M \rangle a)^\circ &= e[(M)^\circ], & (e[K])^\circ &= \langle (K)^\circ \rangle a, \\ (\langle M \rangle e)^\circ &= a[(M)^\circ], & (a[K])^\circ &= \langle (K)^\circ \rangle e, \\ (\forall R)^\circ &= (\exists L), & (\exists L)^\circ &= (\forall R), & (\forall L)^\circ &= (\exists R), & (\exists R)^\circ &= (\forall L). \end{aligned}$$

This duality transformation preserves substitution of types, terms, and coterms.

Lemma 5.10. *Let A and B be types, D be an expression, M be a term, and K be a coterms of DC2. Then the following hold.*

- (1) $(A[B/X])^\circ = (A)^\circ[(B)^\circ/X]$.
- (2) $(D[M/x])^\circ = (D)^\circ[(M)^\circ/x']$.
- (3) $(D[K/\alpha])^\circ = (D)^\circ[(K)^\circ/\alpha']$.

Proof. The claim (1) is shown by induction on A . The claims (2) and (3) are shown by induction on D . \square

The extended duality transformation preserves typing and reduction. It is an involution in DC2.

Proposition 5.11 (Duality of DC2). *The following claims hold.*

- (1) *If J is derived from J_1, \dots, J_n ($n = 1$ or 2) by an inference rule R , then $(J)^\circ$ is derived from $(J_n)^\circ, \dots, (J_1)^\circ$ by the inference rule $(R)^\circ$.*
- (2) *$D \longrightarrow_{\text{DC2}} E$ implies $(D)^\circ \longrightarrow_{\text{DC2}} (E)^\circ$.*
- (3) *$((A)^\circ)^\circ = A$, $((D)^\circ)^\circ = D$, and $((J)^\circ)^\circ = J$ hold.*

Proof. The claim (1) is proved by case analysis of the inference rules of DC2. The claim (2) is proved by induction of the generation of $\longrightarrow_{\text{DC2}}$ using Lemma 5.10. The claim (3) is proved by induction on types and expressions. \square

Next we give a definition of the second-order symmetric λ -calculus $S\lambda 2$. The symmetric λ -calculus is introduced by Barbanera and Berardi [2] as a classical extension of the λ -calculus. The strong normalization of its second-order extension $S\lambda 2$ is proved by Parigot [21] using the reducibility method. The particular system we consider here is an extension of Parigot's system with two additional rules (η_r and η_l). As discussed in *ibid.*, Parigot's proof works with this variant without problem.

Definition 5.12 ($S\lambda 2$). We define the second-order symmetric λ -calculus $S\lambda 2$. The types of $S\lambda 2$ are either the special type \perp or m-types (denoted by τ, σ, \dots) given by:

$$\tau ::= X \mid X^\perp \mid \tau \times \tau \mid \tau + \tau \mid \forall X. \tau \mid \exists X. \tau$$

where X, Y, \dots range over type variables. The types $\forall X. \tau$ and $\exists X. \tau$ bind X in τ . The negation $(\tau)^\perp$ of τ is defined by:

$$\begin{aligned} (X)^\perp &= X^\perp, & (X^\perp)^\perp &= X, \\ (\tau \times \sigma)^\perp &= (\tau)^\perp + (\sigma)^\perp, & (\tau + \sigma)^\perp &= (\tau)^\perp \times (\sigma)^\perp, \\ (\forall X. \tau)^\perp &= \exists X. (\tau)^\perp, & (\exists X. \tau)^\perp &= \forall X. (\tau)^\perp. \end{aligned}$$

The symbols $x, y, \dots, \alpha, \beta, \dots$ range over variables. The terms of $S\lambda 2$, denoted by t, u, \dots , are defined by

$$t ::= x \mid \text{inj}_1(t) \mid \text{inj}_2(t) \mid \langle t, t \rangle \mid t * t \mid \lambda x. t \mid \mathbf{a}(t) \mid \mathbf{e}(t).$$

The one-step reduction relation $\longrightarrow_{S\lambda 2}$ of $S\lambda 2$ is defined as the compatible closure of the following rules:

$$\begin{array}{ll} (\beta_r) & (\lambda x. t) * u \longrightarrow_{S\lambda 2} t[u/x], & (\beta_l) & u * (\lambda x. t) \longrightarrow_{S\lambda 2} t[u/x], \\ (\beta_{\times+1}) & \langle t_1, t_2 \rangle * \text{inj}_1(u) \longrightarrow_{S\lambda 2} t_1 * u, & (\beta_{\times+1}) & \text{inj}_1(u) * \langle t_1, t_2 \rangle \longrightarrow_{S\lambda 2} u * t_1, \\ (\beta_{\times+2}) & \langle t_1, t_2 \rangle * \text{inj}_2(u) \longrightarrow_{S\lambda 2} t_2 * u, & (\beta_{\times+2}) & \text{inj}_2(u) * \langle t_1, t_2 \rangle \longrightarrow_{S\lambda 2} u * t_2, \\ (\beta_{\forall\exists}) & \mathbf{a}(t) * \mathbf{e}(u) \longrightarrow_{S\lambda 2} t * u, & (\beta_{\forall\exists}) & \mathbf{e}(u) * \mathbf{a}(t) \longrightarrow_{S\lambda 2} u * t, \\ (\eta_r) & \lambda y. (y * t) \longrightarrow_{S\lambda 2} t, & (\eta_l) & \lambda y. (t * y) \longrightarrow_{S\lambda 2} t, \end{array}$$

where y is not free in t in (η_l) and (η_r) .

A typing context (denoted by Γ, Δ) is a finite set and of the form $x_1 : \tau_1, \dots, x_n : \tau_n$. A judgment of $S\lambda 2$ takes either the form $\Gamma \vdash t : \tau$ or $\Gamma \vdash t : \perp$. The typing rules of $S\lambda 2$ are defined as follows:

$$\begin{array}{c} \frac{}{\Gamma, x : \tau \vdash x : \tau} (\text{Ax}) \quad \frac{\Gamma, x : \tau \vdash t : \perp}{\Gamma \vdash \lambda x. t : (\tau)^\perp} (\text{abs}) \quad \frac{\Gamma \vdash t : (\tau)^\perp \quad \Gamma \vdash u : \tau}{\Gamma \vdash t * u : \perp} (\text{app}) \\[10pt] \frac{\Gamma \vdash t : \tau_i}{\Gamma \vdash \text{inj}_i(t) : \tau_1 + \tau_2} (+_i) \ (i = 1, 2) \quad \frac{\Gamma \vdash t : \tau \quad \Gamma \vdash u : \sigma}{\Gamma \vdash \langle t, u \rangle : \tau \times \sigma} (\times) \\[10pt] \frac{\Gamma \vdash t : \tau}{\Gamma \vdash a(t) : \forall X. \tau} (\forall) \quad (X \text{ is not free in } \Gamma) \quad \frac{\Gamma \vdash t : \tau[\sigma/X]}{\Gamma \vdash e(t) : \exists X. \tau} (\exists) \end{array}$$

Theorem 5.13 (Strong normalization of $S\lambda 2$ [21]). *Every typable term is strongly normalizing in $S\lambda 2$.*

We will give a reduction-preserving and type-preserving translation from DC2 into $S\lambda 2$. Our translation is a second-order extension of the translation from DC into the symmetric λ -calculus given by Tzevelekos [26].

Definition 5.14. Let A be a type of DC2. The type $(A)^\dagger$ of $S\lambda 2$ is defined as follows:

$$\begin{aligned} (X)^\dagger &= X, & (A \wedge B)^\dagger &= (A)^\dagger \times (B)^\dagger, & (A \vee B)^\dagger &= (A)^\dagger + (B)^\dagger, \\ (\neg A)^\dagger &= ((A)^\dagger)^\perp, & (\forall X. A)^\dagger &= \forall X. (A)^\dagger, & (\exists X. A)^\dagger &= \exists X. (A)^\dagger. \end{aligned}$$

Let D be an expression of DC2. The term $(D)^\dagger$ of $S\lambda 2$ is defined by:

$$\begin{aligned} (x)^\dagger &= x, & (\alpha)^\dagger &= \alpha, \\ ((S).\alpha)^\dagger &= \lambda \alpha. (S)^\dagger, & (x.(S))^\dagger &= \lambda x. (S)^\dagger, \\ (\langle M \rangle a)^\dagger &= a((M)^\dagger), & (\langle M \rangle e)^\dagger &= e((M)^\dagger), \\ (e[K])^\dagger &= a((K)^\dagger), & (a[K])^\dagger &= e((K)^\dagger), \\ (\langle M \rangle \text{inl})^\dagger &= \text{inj}_1((M)^\dagger), & (\text{fst}[K])^\dagger &= \text{inj}_1((K)^\dagger), \\ (\langle M \rangle \text{inr})^\dagger &= \text{inj}_2((M)^\dagger), & (\text{snd}[K])^\dagger &= \text{inj}_2((K)^\dagger), \\ (\langle M, N \rangle)^\dagger &= \langle (M)^\dagger, (N)^\dagger \rangle, & ([K, L])^\dagger &= \langle (K)^\dagger, (L)^\dagger \rangle, \\ ([K] \text{not})^\dagger &= \lambda x. (x * (K)^\dagger), & (\text{not} \langle M \rangle)^\dagger &= (M)^\dagger, \\ (M \bullet K)^\dagger &= (M)^\dagger * (K)^\dagger. \end{aligned}$$

We define the translation of $[K] \text{not}$ by using η -expansion, so that all reductions in DC2 are strictly simulated in $S\lambda 2$.

$(\Gamma)^\dagger$ and $((\Delta)^\dagger)^\perp$ are defined as $x_1 : (A_1)^\dagger, \dots, x_n : (A_n)^\dagger$ and $\alpha_1 : ((B_1)^\dagger)^\perp, \dots, \alpha_m : ((B_m)^\dagger)^\perp$ respectively if Γ is $x_1 : A_1, \dots, x_n : A_n$, and Δ is $\alpha_1 : B_1, \dots, \alpha_m : B_m$. For a judgment J of DC2, the judgment $(J)^\dagger$ of $S\lambda 2$ is defined as follows: The judgment $(\Gamma \vdash \Delta \mid M : A)^\dagger$ is defined as $(\Gamma)^\dagger, ((\Delta)^\dagger)^\perp \vdash (M)^\dagger : (A)^\dagger$. The judgment $(K : A \mid \Gamma \vdash \Delta)^\dagger$ is defined as $(\Gamma)^\dagger, ((\Delta)^\dagger)^\perp \vdash (K)^\dagger : ((A)^\dagger)^\perp$. The judgment $(\Gamma \mid S \vdash \Delta)^\dagger$ is defined as $(\Gamma)^\dagger, ((\Delta)^\dagger)^\perp \vdash (S)^\dagger : \perp$.

This translation preserves provability and one-step reductions.

Proposition 5.15. *The following claims hold.*

- (1) *If J is provable in DC2, then $(J)^\dagger$ is provable in $S\lambda 2$.*
- (2) *$D \longrightarrow_{\text{DC2}} E$ implies $(D)^\dagger \longrightarrow_{S\lambda 2} (E)^\dagger$.*

Proof. The claim (1) is shown by induction on the proof of J . The claim (2) is shown by induction on the definition of $\longrightarrow_{\text{DC2}}$. \square

We can obtain strong normalization of DC2 from the above proposition.

Theorem 5.16 (Strong normalization of DC2). *Every typable expression is strongly normalizing in DC2.*

Proof. Assume there is an infinite reduction sequence

$$D = D_0 \longrightarrow_{\text{DC2}} D_1 \longrightarrow_{\text{DC2}} \dots$$

starting from D . From Proposition 5.15, the expression $(D)^\dagger$ is typable in $S\lambda 2$, and

$$(D)^\dagger \longrightarrow_{S\lambda 2} (D_1)^\dagger \longrightarrow_{S\lambda 2} \dots$$

is an infinite reduction sequence. This contradicts Theorem 5.13. \square

Remark 5.17. Tzevelekos [26] also gave a back translation $(-)^p$ from the symmetric λ -calculus into DC. As noted in his paper (Section 3, Note 3.5), this translation cannot extend to the second-order case since it does not preserve typing judgments for existential quantification. This is because the translation $(-)^p$ does not preserve type substitution: $(A[B/X])^p \neq (A)^p[(B)^p/X]$. The same argument applies to ours.

6. STRONG NORMALIZATION OF $\text{DC}\mu\nu$

In this section, we prove strong normalization in $\text{DC}\mu\nu$. We will give a translation from $\text{DC}\mu\nu$ into DC2 that is based on the second-order encoding of inductive and coinductive types. Our proof of strong normalization will be done by showing the fact that one-step reduction in $\text{DC}\mu\nu$ is translated to one or more steps reduction in DC2.

We use the following degree of expressions in $\text{DC}\mu\nu$ for defining the second-order coding of inductive and coinductive types.

When we try to prove some properties of expressions by induction on expressions, that induction sometimes does not work, since the expression contains $\text{mono}_{A,B,x.M}^{X,C}\{N\}$ that is defined by using induction on $\|C\|_X$. In order for solving this, we will introduce the pair of the summation of $\|C\|_X$ and the size of an expression as a measure.

Definition 6.1. Let D be an expression in $\text{DC}\mu\nu$. The number $\|D\|$ is defined by:

$$\begin{aligned} \|x\| &= \|\alpha\| = 0, \\ \|\langle M, N \rangle\| &= \|\text{coitr}_x^A\langle M, N \rangle\| = \max(\|M\|, \|N\|), \\ \|M \bullet K\| &= \max(\|M\|, \|K\|), \\ \|[K, L]\| &= \|\text{itr}_\alpha^A[K, L]\| = \max(\|K\|, \|L\|), \\ \|(S).\alpha\| &= \|x.(S)\| = \|S\|, \\ \|\langle M \rangle \text{inl}\| &= \|\langle M \rangle \text{inr}\| = \|\text{not}\langle M \rangle\| = \|M\|, \\ \|\text{in}^{\mu X.A}\langle M \rangle\| &= \|M\| + \|A\|_X + 1, \\ \|\text{fst}[K]\| &= \|\text{snd}[K]\| = \|[K] \text{not}\| = \|K\|, \\ \|\text{out}^{\nu X.A}[K]\| &= \|K\| + \|A\|_X + 1. \end{aligned}$$

The number $|D|$ is defined by:

$$\begin{aligned}
|x| &= |\alpha| = 0, \\
|\langle M, N \rangle| &= |\text{coitr}_x^A \langle M, N \rangle| = |M| + |N| + 1, \\
|M \bullet K| &= |M| + |K| + 1, \\
|[K, L]| &= |\text{itr}_\alpha^A [K, L]| = |K| + |L| + 1, \\
|(S).\alpha| &= |x.(S)| = |S| + 1, \\
|\langle M \rangle \text{inl}| &= |\langle M \rangle \text{inr}| = |\text{not} \langle M \rangle| = |\text{in}^{\mu X, A} \langle M \rangle| = |M| + 1, \\
|\text{fst}[K]| &= |\text{snd}[K]| = |[K] \text{not}| = |\text{out}^{\nu X, A} [K]| = |K| + 1.
\end{aligned}$$

The degree $\deg(D)$ of the expression D is defined as the pair $(\|D\|, |D|)$. We also define the order of the degrees by the lexicographic order.

The number $|D|$ is the number of constructors in the expression D . The number $\|D\|$ is the maximum summation of $(\|A\|_X + 1)$ for $\text{in}^{\mu X, A} \langle M \rangle$ and $\text{out}^{\nu X, A} [K]$ in paths in D . For example, $\deg(\text{in}^{\mu X, \neg X \vee X} \langle \langle [x.(\langle x \rangle \text{inr} \bullet \gamma)] \text{not} \rangle \text{inl} \bullet \gamma \rangle) = (4, 7)$. We have $\|E\| \leq \|D\|$ and $|E| < |D|$ when the expression E is a proper subexpression of D . The degree satisfies the following properties.

Lemma 6.2. *The following claims hold.*

- (1) $\|D\| = \|(D)^\circ\|$ and $|D| = |(D)^\circ|$ hold.
- (2) $\|\text{mono}_{B, C, x}^{X, A} \{M, N\}\| \leq \|M\| + \|N\| + \|A\|_X$ holds.
- (3) $\deg(\text{in}^{\mu X, A} \langle M \rangle) > \deg(\text{mono}_{\mu X, A, Y, \alpha}^{X, A} \{x.(y \bullet (x @ \alpha)), \beta\})$ holds.

Proof. The claims of (1) are shown by induction on D . The claim (2) is shown by induction on $\|A\|_X$. The claim (3) is proved by using (2). \square

We present the second-order encoding for $\text{DC}\mu\nu$. We will write $\lambda(x, \alpha).S$ for $\lambda x.((S).\alpha)$. Then $(\lambda(x, \alpha).S) \bullet (N @ K)$ is reduced to $S[N/x][K/\alpha]$.

Definition 6.3 (Translation $\overline{(-)}$ from $\text{DC}\mu\nu$ into DC2). Let A be a type of $\text{DC}\mu\nu$. The type \overline{A} of DC2 is defined as follows:

$$\begin{aligned}
\overline{X} &= X, & \overline{A \wedge B} &= \overline{A} \wedge \overline{B}, & \overline{\neg A} &= \neg \overline{A}, & \overline{A \vee B} &= \overline{A} \vee \overline{B}, \\
\overline{\mu X. A} &= \forall X. (\overline{A} \supset X) \supset X, & \overline{\nu X. A} &= \exists X. (\neg(\neg \overline{A} \wedge X) \wedge X),
\end{aligned}$$

where \supset is defined in Definition 2.9. For an expression D of $\text{DC}\mu\nu$, the expression \overline{D} of DC2 is defined by induction on $\deg(D)$ as follows. For the expressions D of the same degree, we first define \overline{D} for D such that D is not of the form $\overline{\text{out}^{\nu X, A} [K]}$ or $\overline{\text{coitr}_x^A \langle M, N \rangle}$, and we next define \overline{D} for D such that D is of the form $\overline{\text{out}^{\nu X, A} [K]}$ or $\overline{\text{coitr}_x^A \langle M, N \rangle}$.

$$\begin{aligned}
\overline{x} &= x, & \overline{\alpha} &= \alpha, \\
\overline{(S).\alpha} &= \overline{(S)}.\alpha, & \overline{x.(S)} &= x.(\overline{S}), \\
\overline{\langle M, N \rangle} &= \langle \overline{M}, \overline{N} \rangle, & \overline{[K, L]} &= [\overline{K}, \overline{L}], \\
\overline{\langle M \rangle \text{inl}} &= \langle \overline{M} \rangle \text{inl}, & \overline{\text{fst}[K]} &= \text{fst}[\overline{K}], \\
\overline{\langle M \rangle \text{inr}} &= \langle \overline{M} \rangle \text{inr}, & \overline{\text{snd}[K]} &= \text{snd}[\overline{K}], \\
\overline{[K] \text{not}} &= [\overline{K}] \text{not}, & \overline{\text{not} \langle M \rangle} &= \text{not} \langle \overline{M} \rangle,
\end{aligned}$$

$$\begin{aligned}
\overline{M \bullet K} &= \overline{M} \bullet \overline{K}, \\
\overline{\text{itr}_\alpha^A[K, L]} &= \mathbf{a}[(\lambda(x, \alpha).(x \bullet \overline{K}))@ \overline{L}], \\
\overline{\text{out}^{\nu X.A}[K]} &= (\overline{\text{in}^{\mu X.(A)^\circ} \langle (K)^\circ \rangle})^\circ, \\
\overline{\text{in}^{\mu X.A} \langle M \rangle} &= \langle \lambda(y, \beta).(y \bullet ((Q_Y[X.A] \bullet \mathcal{R}_M\{y, \gamma\}).\gamma @ \beta)) \rangle \mathbf{a}, \\
\overline{\text{coitr}_x^A \langle M, N \rangle} &= (\overline{\text{itr}_{x'}^{(A)^\circ} [(M)^\circ, (N)^\circ]})^\circ,
\end{aligned}$$

where $Q_Y[X.A]$ is defined as $\lambda y. \lambda(z, \beta). (z \bullet \text{mono}_{\mu X.A, Y, \alpha}^{X.A} \{x.(y \bullet (x @ \alpha)), \beta\})$, and $\mathcal{R}_M\{N, K\}$ is defined as $(\lambda(x, \alpha).(x \bullet \mathbf{a}[N @ \alpha])) @ (\overline{M} @ K)$.

We also define the translation of judgments. The context $\overline{\Gamma}$ is defined as $x_1 : \overline{A_1}, \dots, x_n : \overline{A_n}$ if Γ is $x_1 : A_1, \dots, x_n : A_n$. The cocontext $\overline{\Delta}$ is defined as $\alpha_1 : \overline{B_1}, \dots, \alpha_m : \overline{B_m}$ if Δ is $\alpha_1 : B_1, \dots, \alpha_m : B_m$. The judgment $\overline{\Gamma \vdash \Delta \mid M : A}$ is defined as $\overline{\Gamma} \vdash \overline{\Delta} \mid \overline{M} : \overline{A}$. The judgment $\overline{K : A \mid \Gamma \vdash \Delta}$ is defined as $\overline{K} : \overline{A} \mid \overline{\Gamma} \vdash \overline{\Delta}$. The judgment $\overline{\Gamma \mid S \vdash \Delta}$ is defined as $\overline{\Gamma} \mid \overline{S} \vdash \overline{\Delta}$.

The next lemma shows that this translation commutes with $(-)^\circ$.

Lemma 6.4. $(\overline{A})^\circ = \overline{(A)^\circ}$, $(\overline{D})^\circ = \overline{(D)^\circ}$, and $(\overline{J})^\circ = \overline{(J)^\circ}$ hold.

Proof. The claim for A is proved by induction on A .

The claim for D is proved by induction on D . The cases of $\overline{\text{in}^{\mu X.A} \langle M \rangle}$, $\overline{\text{out}^{\nu X.A}[K]}$, $\overline{\text{itr}_\alpha^A[K, L]}$, and $\overline{\text{coitr}_x^A \langle M, N \rangle}$ are shown by the definition of the translation and the dualities of $\text{DC}\mu\nu$ and DC2 . The case of $\overline{\text{in}^{\mu X.A} \langle M \rangle}$ is shown as follows: $(\overline{\text{in}^{\mu X.A} \langle M \rangle})^\circ = (\overline{\text{in}^{\mu X.((A)^\circ)^\circ} \langle ((M)^\circ)^\circ \rangle})^\circ = \overline{\text{out}^{\nu X.(A)^\circ} [(M)^\circ]} = \overline{(\text{in}^{\mu X.A} \langle M \rangle)^\circ}$. We can show the cases $\overline{\text{out}^{\nu X.A}[K]}$, $\overline{\text{itr}_\alpha^A[K, L]}$, and $\overline{\text{coitr}_x^A \langle M, N \rangle}$ similarly. The other cases are straightforwardly proved by the induction hypothesis.

The claim for J is proved by using the claims for A and D . □

The translation $\overline{(-)}$ preserves substitution.

Lemma 6.5. $\overline{A[B/X]} = \overline{A}[\overline{B}/\overline{X}]$, $\overline{D[N/x]} = \overline{D}[\overline{N}/\overline{x}]$, and $\overline{D[L/\alpha]} = \overline{D}[\overline{L}/\overline{\alpha}]$ hold.

Proof. The first claim is shown by induction on A . The second and the third claims are shown simultaneously by induction on $\deg(D)$. For the expressions \overline{D} of the same degree, we first show the claims for \overline{D} such that D is not of the form $\overline{\text{out}^{\nu X.A}[K]}$ or $\overline{\text{coitr}_x^A \langle M, N \rangle}$, and we next show the claims for \overline{D} such that D is of the form $\overline{\text{out}^{\nu X.A}[K]}$ or $\overline{\text{coitr}_x^A \langle M, N \rangle}$.

We consider the cases of $\overline{\text{in}^{\mu X.A} \langle M \rangle}$, $\overline{\text{itr}_\alpha^A[K, L]}$, $\overline{\text{out}^{\nu X.A}[K]}$ and $\overline{\text{coitr}_x^A \langle M, N \rangle}$. The other cases are straightforwardly proved by the induction hypothesis.

The second claim of the case $\overline{\text{in}^{\mu X.A} \langle M \rangle}$ is shown in the following way. By the induction hypothesis, we have $\text{mono}_{\mu X.A, Y, \alpha}^{X.A} \{x.(y \bullet (x @ \alpha)), \beta\}[\overline{N}/x] = \text{mono}_{\mu X.A, Y, \alpha}^{X.A} \{x.(y \bullet (x @ \alpha)), \beta\}[N/x]$ since $\deg(\text{mono}_{\mu X.A, Y, \alpha}^{X.A} \{x.(y \bullet (x @ \alpha)), \beta\}) < \deg(\text{in}^{\mu X.A} \langle M \rangle)$ by Lemma 6.2 (3). By Lemma 3.13 (2a), $\text{mono}_{\mu X.A, Y, \alpha}^{X.A} \{x.(y \bullet (x @ \alpha)), \beta\}[N/x] = \text{mono}_{\mu X.A, Y, \alpha}^{X.A} \{x.(y \bullet (x @ \alpha)), \beta\}$. Hence we have $(Q_Y[X.A])[\overline{N}/x] = Q_Y[X.A]$. By the induction hypothesis, we have $\mathcal{R}_M\{y, \gamma\}[\overline{N}/x] = \mathcal{R}_{M[N/x]}\{y, \gamma\}$ since $\deg(M) < \deg(\text{in}^{\mu X.A} \langle M \rangle)$. Therefore, $\overline{\text{in}^{\mu X.A} \langle M \rangle}[\overline{N}/x]$ is equal to $\langle \lambda(y, \beta).(y \bullet ((Q_Y[X.A])[\overline{N}/x] \bullet (\mathcal{R}_M\{y, \gamma\})[\overline{N}/x].\gamma @ \beta)) \rangle \mathbf{a}$. Then it is equal to $\langle \lambda(y, \beta).(y \bullet ((Q_Y[X.A] \bullet \mathcal{R}_{M[N/x]}\{y, \gamma\}).\gamma @ \beta)) \rangle \mathbf{a}$. The last term is equal to $(\text{in}^{\mu X.A} \langle M \rangle)[N/x]$ by the definition of $\overline{(-)}$.

The second claim of the case $\text{itr}_\alpha^A[K, L]$ is shown in the following way. The coterm $\text{itr}_\alpha^A[K, L][\overline{N}/x]$ is equal to $\mathbf{a}[(\lambda(y, \alpha). (y \bullet \overline{K}[\overline{N}/x])) @ \overline{L}[\overline{N}/x]]$. By the induction hypothesis, it is equal to $\mathbf{a}[(\lambda(y, \alpha). (y \bullet \overline{K}[\overline{N}/x])) @ \overline{L}[\overline{N}/x]]$. Hence it is equal to $(\text{itr}_\alpha^A[K, L])[\overline{N}/x]$ by the definition of $(-)$.

The second claim of the case $\text{out}^{\nu X.A}[K]$ is shown in the following way. Since $\|K\| = \|(K)^\circ\|$ and $|K| = |(K)^\circ|$ by Lemma 6.2 (1), we have $\deg(\text{out}^{\nu X.A}[K]) = \deg(\text{in}^{\mu X.(A)^\circ} \langle (K)^\circ \rangle)$. Hence $\text{in}^{\mu X.(A)^\circ} \langle (K[\overline{N}/x])^\circ \rangle = \text{in}^{\mu X.(A)^\circ} \langle (K)^\circ \rangle [(\overline{N})^\circ / x']$ holds by Lemma 3.10, since the third claim for $\text{in}^{\mu X.(A)^\circ} \langle (K)^\circ \rangle$ is already shown before this case. Then we can obtain the claim of this case as follows:

$$\begin{aligned} \overline{(\text{out}^{\nu X.A}[K])[\overline{N}/x]} &= \overline{(\text{in}^{\mu X.(A)^\circ} \langle (K[\overline{N}/x])^\circ \rangle)^\circ} = \overline{(\text{in}^{\mu X.(A)^\circ} \langle (K)^\circ \rangle [(\overline{N})^\circ / x'])^\circ} = \\ &= (\text{in}^{\mu X.(A)^\circ} \langle (K)^\circ \rangle)^\circ [((\overline{N})^\circ)^\circ / x] = \text{out}^{\nu X.A}[K][(\overline{N})^\circ / x] = \text{out}^{\nu X.A}[K][\overline{N}/x]. \end{aligned}$$

The third claim of this case is shown similarly.

The second and third claims of the case $\text{coitr}_x^A\langle M, N \rangle$ is shown in the similar way to the case of $\text{out}^{\nu X.A}[K]$. \square

Note that the second and third claims of the above lemma cannot be proved straightforwardly by induction on D . For example, for proving the case of $\text{in}^{\mu X.A}\langle M \rangle$ in the second claim, we need induction hypothesis for $\text{mono}_{\mu X.A, Y, \alpha}^{X.A} \{x.(y \bullet (x @ \alpha)), \beta\}$ but it is not a subterm of $\text{in}^{\mu X.A}\langle M \rangle$.

The next proposition says the translation $(-)$ preserves provability.

Proposition 6.6. *If J is provable in $\text{DC}\mu\nu$, then \overline{J} is provable in DC2 .*

Proof. This is shown by induction on the degree of the principal expression in J . We show the cases of $\text{in}^{\mu X.A}\langle M \rangle$, $\text{out}^{\nu X.A}[K]$, $\text{itr}_\alpha^A[K, L]$, and $\text{coitr}_x^A\langle M, N \rangle$.

The case of $\text{itr}_\alpha^A[K, L]$ is shown by the induction hypothesis and Lemma 6.5. The cases of $\text{coitr}_x^A\langle M, N \rangle$ and $\text{out}^{\nu X.A}[K]$ are shown by the induction hypothesis and the dualities of $\text{DC}\mu\nu$ and DC2 .

We prove the case of $\text{in}^\mu\langle M \rangle$. We write μ , $A[B]$, and $\overline{A}[C]$ as abbreviations of $\mu X.A$, $A[B/X]$, and $\overline{A}[C/X]$ respectively. This case is shown by the following three steps: (a) we show $\mathcal{R}_M\{y, \gamma\} : (\overline{\mu} \supset Y) \supset \overline{A}[\overline{\mu}] \supset \overline{A}[Y] \mid \overline{\Gamma}, y : \overline{A}[Y] \supset Y \vdash \overline{\Delta}, \gamma : \overline{A}[Y]$ is derivable, where $\mathcal{R}_M\{y, \gamma\}$ is $(\lambda(x, \alpha).(x \bullet \mathbf{a}[y @ \alpha])) @ (\overline{M} @ \gamma)$. Next, (b) we show $\vdash \mid Q_Y[X.A] : (\overline{\mu} \supset Y) \supset \overline{A}[\overline{\mu}] \supset \overline{A}[Y]$ is derivable, where $Q_Y[X.A]$ is $\lambda y. \lambda(z, \beta).(z \bullet \text{mono}_{\mu X.A, Y, \alpha}^{X.A} \{x.(y \bullet (x @ \alpha)), \beta\})$. Finally, (c) we can easily show $\overline{\Gamma} \vdash \overline{\Delta} \mid \text{in}^\mu\langle M \rangle : \mu$ from (a) and (b).

The claim (a) is shown in the following way. Suppose $\Gamma \vdash \Delta \mid \text{in}^\mu\langle M \rangle : \mu$ is derivable. Then we have the derivation of $\Gamma \vdash \Delta \mid M : A[\mu]$. By the induction hypothesis and Lemma 6.5, we obtain $\overline{\Gamma} \vdash \overline{\Delta} \mid \overline{M} : \overline{A}[\overline{\mu}]$. Then we have a derivation of $\overline{M} @ \gamma : \overline{A}[\overline{\mu}] \supset \overline{A}[Y] \mid \overline{\Gamma} \vdash \overline{\Delta}, \gamma : \overline{A}[Y]$ by $(\supset L)$ rule. On the other hand, we can show $y : \overline{A}[Y] \supset Y \vdash \mid \lambda(x, \alpha).(x \bullet \mathbf{a}[y @ \alpha]) : \overline{\mu} \supset Y$. Then we have $\mathcal{R}_M\{y, \gamma\} : (\overline{\mu} \supset Y) \supset \overline{A}[\overline{\mu}] \supset \overline{A}[Y] \mid \overline{\Gamma}, y : \overline{A}[Y] \supset Y \vdash \overline{\Delta}, \gamma : \overline{A}[Y]$.

The claim (b) is shown as follows. We can show $\text{mono}_{\mu, Y, \alpha}^{X.A} \{x.(y \bullet (x @ \alpha)), \beta\} : A[\mu] \mid y : \mu \supset Y \vdash \beta : A[Y]$ in $\text{DC}\mu\nu$ by using Lemma 3.14, the judgment $x.(y \bullet (x @ \alpha)) : \mu \mid y : \mu \supset Y \vdash \alpha : Y, \beta : A[Y]$, and $\beta : A[\mu] \mid y : \mu \supset Y \vdash \beta : A[Y]$. By Lemma 6.2 (3), we have $\deg(\text{in}^{\mu X.A}\langle M \rangle) > \deg(\text{mono}_{\mu, Y, \alpha}^{X.A} \{x.(y \bullet (x @ \alpha)), \beta\})$. Hence $\text{mono}_{\mu, Y, \alpha}^{X.A} \{x.(y \bullet (x @ \alpha)), \beta\} : \overline{A}[\overline{\mu}] \mid y : \overline{\mu} \supset Y \vdash \beta : \overline{A}[Y]$ is derivable by induction hypothesis and Lemma 6.5. Therefore we obtain $\vdash \mid Q_Y[X.A] : (\overline{\mu} \supset Y) \supset \overline{A}[\overline{\mu}] \supset \overline{A}[Y]$.

The other cases are straightforwardly proved by the induction hypothesis. \square

The translation $\overline{(-)}$ maps one-step reduction to one or more steps of reduction.

Proposition 6.7. *For expressions D and E of $\text{DC}\mu\nu$, the relation $D \longrightarrow_{\text{DC}\mu\nu} E$ implies $\overline{D} \longrightarrow_{\text{DC2}}^+ \overline{E}$.*

Proof. First we show the claim without $(\beta\mu)$ nor $(\beta\nu)$ by induction on $\longrightarrow_{\text{DC}\mu\nu}$ with Lemma 6.5. Next, by using this and Lemma 6.5, we show the claim of this proposition by induction on $\longrightarrow_{\text{DC}\mu\nu}$. We consider cases according to the reduction rule.

The case of $(\beta\mu)$ is shown as follows: Suppose we have $\overline{\text{in}^{\mu X.A}\langle M \rangle \bullet \text{itr}_\alpha^B[K, L]}$. This is equal to $\langle \lambda(y, \beta). (y \bullet ((Q_Y[X.A] \bullet \mathcal{R}_M\{y, \gamma\}).\gamma @ \beta)) \mathbf{a} \bullet \mathbf{a}[(\lambda(x, \alpha).(x \bullet \overline{K})) @ \overline{L}] \rangle$. It is reduced to $\langle \lambda(y, \beta). (y \bullet ((Q_Y[X.A] \bullet \mathcal{R}_M\{y, \gamma\}).\gamma @ \beta)) \rangle \bullet \langle (\lambda(x, \alpha).(x \bullet \overline{K})) @ \overline{L} \rangle$, and then we have $\langle \lambda(x, \alpha).(x \bullet \overline{K}) \rangle \bullet \langle (Q_Y[X.A] \bullet \mathcal{R}_M\{\lambda(x, \alpha).(x \bullet \overline{K}), \gamma\}).\gamma @ \overline{L} \rangle$ by more than one step reduction. Since $\lambda(x, \alpha).(x \bullet \overline{K})$ equals $\lambda x.((x \bullet \overline{K}).\alpha)$, we have $\langle (Q_Y[X.A] \bullet \mathcal{R}_M\{\lambda(x, \alpha).(x \bullet \overline{K}), \gamma\}).\gamma @ \overline{L} \rangle$ by $(\beta \supset)$. This is reduced to $\langle (Q_Y[X.A] \bullet \mathcal{R}_M\{\lambda(x, \alpha).(x \bullet \overline{K}), \overline{K}\}).\alpha @ \overline{L} \rangle$ by (βR) . Here $\mathcal{R}_M\{\lambda(x, \alpha).(x \bullet \overline{K}), \overline{K}\}$ is equal to $\langle \lambda(y, \beta). (y \bullet \text{itr}_\alpha^B[K, \beta]) @ (\overline{M} @ \overline{K}) \rangle$. Hence we can reduce $Q_Y[X.A] \bullet \mathcal{R}_M\{\lambda(x, \alpha).(x \bullet \overline{K}), \overline{K}\}$ to $\overline{M} \bullet \text{mono}_{\mu, Y, \alpha}^{X.A}\{\text{itr}_\alpha^B[K, \beta], K\}$ by using Lemma 6.5 and the first claim. Therefore the previously obtained expression $\langle (Q_Y[X.A] \bullet \mathcal{R}_M\{\lambda(x, \alpha).(x \bullet \overline{K}), \overline{K}\}).\alpha @ \overline{L} \rangle$ is reduced to $\langle \overline{M} \bullet \text{mono}_{\mu X.A, Y, \alpha}^{X.A}\{\text{itr}_\alpha^B[K, \beta], K\} \rangle \bullet \overline{L}$. This is equal to $\langle \overline{M} \bullet \text{mono}_{\mu X.A, Y, \alpha}^{X.A}\{\text{itr}_\alpha^B[K, \beta], K\} \rangle \bullet \overline{L}$.

The case of $(\beta\nu)$ is shown by using the duality of $(\beta\nu)$ and $(\beta\mu)$, the duality of DC2, and Lemma 6.4.

Other cases are shown straightforwardly. \square

Finally, we complete a proof of strong normalization of $\text{DC}\mu\nu$.

Theorem 6.8 (Strong normalization of $\text{DC}\mu\nu$). *Every typable expression of $\text{DC}\mu\nu$ is strongly normalizing.*

Proof. Assume that D is typable in $\text{DC}\mu\nu$ and there is an infinite reduction sequence

$$D \longrightarrow_{\text{DC}\mu\nu} D_1 \longrightarrow_{\text{DC}\mu\nu} \dots$$

starting from D . Then \overline{D} is typable in DC2 by Proposition 6.6 and

$$\overline{D} \longrightarrow_{\text{DC2}}^+ \overline{D}_1 \longrightarrow_{\text{DC2}}^+ \dots$$

is an infinite reduction sequence starting from \overline{D} by Proposition 6.7. This contradicts Theorem 5.16. \square

7. THE CALL-BY-VALUE AND CALL-BY-NAME $\text{DC}\mu\nu$

The motivation for introducing the dual calculus in [27] was to show the duality between call-by-value and call-by-name. In this section, we follow this motivation. That is, we will extend the duality to inductive and coinductive types by introducing the call-by-value and call-by-name variants of $\text{DC}\mu\nu$. These variants also satisfy the important properties such as strong normalization and the Church-Rosser property.

We recall the definition of the call-by-value and call-by-name DC. The call-by-value and call-by-name dual calculus use the notion of values and covalues. They are defined as follows.

Definition 7.1 (Values and covalues of DC [27]). The values (denoted by V, W, \dots) and covalues (denoted by P, Q, \dots) of DC are defined by the following grammar:

$$\begin{aligned} V &::= x \mid \langle V, V \rangle \mid \langle V \rangle \text{inl} \mid \langle V \rangle \text{inr} \mid [K] \text{not}, \\ P &::= \alpha \mid [P, P] \mid \text{fst}[P] \mid \text{snd}[P] \mid \text{not}\langle M \rangle, \end{aligned}$$

where M is a term and K is a cotermin of DC.

The types, expressions, and typing rules of the call-by-value and call-by-name DC are the same as them of DC. The call-by-value reduction relation of DC is defined as follows.

Definition 7.2 (Call-by-value reduction rules of DC). The call-by-value reduction relation $\longrightarrow_{\text{DC}}^v$ of DC is defined from the following rules.

$$\begin{aligned} (\beta \wedge_1)_v & \langle V, W \rangle \bullet \text{fst}[K] \longrightarrow_{\text{DC}}^v V \bullet K, \\ (\beta \wedge_2)_v & \langle V, W \rangle \bullet \text{snd}[K] \longrightarrow_{\text{DC}}^v W \bullet K, \\ (\beta \vee_1)_v & \langle V \rangle \text{inl} \bullet [K, L] \longrightarrow_{\text{DC}}^v V \bullet K, \\ (\beta \vee_2)_v & \langle W \rangle \text{inr} \bullet [K, L] \longrightarrow_{\text{DC}}^v W \bullet L, \\ (\beta \neg)_v & [K] \text{not} \bullet \text{not}\langle M \rangle \longrightarrow_{\text{DC}}^v M \bullet K, \\ (\beta R)_v & (S). \alpha \bullet K \longrightarrow_{\text{DC}}^v S[K/\alpha], \\ (\beta L)_v & V \bullet x.(S) \longrightarrow_{\text{DC}}^v S[V/x], \\ (\varsigma \wedge_1)_v & \langle M, N \rangle \longrightarrow_{\text{DC}}^v (M \bullet x.(\langle x, N \rangle \bullet \alpha)). \alpha, \\ (\varsigma \wedge_2)_v & \langle V, M \rangle \longrightarrow_{\text{DC}}^v (M \bullet x.(\langle V, x \rangle \bullet \alpha)). \alpha, \\ (\varsigma \vee_1)_v & \langle M \rangle \text{inl} \longrightarrow_{\text{DC}}^v (M \bullet x.(\langle x \rangle \text{inl} \bullet \alpha)). \alpha, \\ (\varsigma \vee_2)_v & \langle M \rangle \text{inr} \longrightarrow_{\text{DC}}^v (M \bullet x.(\langle x \rangle \text{inr} \bullet \alpha)). \alpha, \\ (\eta R)_v^+ & M \longrightarrow_{\text{DC}}^v (M \bullet \alpha). \alpha, \quad \text{and} \\ (\eta L)_v^+ & K \longrightarrow_{\text{DC}}^v x.(x \bullet K), \end{aligned}$$

where M is not a value, and x and α in $(\varsigma \wedge_1)_v$, $(\varsigma \wedge_2)_v$, $(\varsigma \vee_1)_v$, $(\varsigma \vee_2)_v$, $(\eta L)_v^+$ and $(\eta R)_v^+$ are fresh.

An example of use of ς -rules is

$$\langle (S). \alpha \rangle \text{inl} \longrightarrow_{\text{DC}}^v ((S). \alpha \bullet x.(\langle x \rangle \text{inl} \bullet \beta)). \beta \longrightarrow_{\text{DC}}^v (S[x.(\langle x \rangle \text{inl} \bullet \beta)/\alpha]). \beta.$$

This system is obtained from the call-by-value dual calculus given in [27] by removing the implication.

We note that the original system in [27] includes implication types, values for implication, and a call-by-value β -rule for implication. However, as mentioned in [27], an implication $A \supset B$ can be defined as $\neg(A \wedge \neg B)$ under call-by-value. Hence each value for implication can be replaced a value in terms of other connectives, and the reduction rule for implication can be simulated by the other β -rules.

The rules $(\varsigma \wedge_1)_v$, $(\varsigma \wedge_2)_v$, $(\varsigma \vee_1)_v$, and $(\varsigma \vee_2)_v$ are the separated forms of the rule (ς) given in [27], and our rules are equivalent to his rule. However, we prefer this separated form since this form is easy to add ς -rules for inductive and coinductive types later.

The symbol $+$ used in $(\eta L)_v^+$ and $(\eta R)_v^+$ means η -expansion rules. When we extend call-by-value and call-by-name calculi with inductive and coinductive types later in this section, we will use the reduction (ηR) and (ηL) instead of the above expansion $(\eta R)_v^+$ and $(\eta L)_v^+$ for the following reasons. In [27], η -rules requires side conditions to avoid infinite reduction sequence: “expansions (ηL) and (ηR) should be applied only to a term M or cotermin K that is not the immediate subject of a cut”. However, two problems still remain about η -expansion rules. One problem is that a value becomes non-value by the η -expansion: For example, a value x is expanded to a non-value $(x \bullet \alpha). \alpha$ by $(\eta R)_v^+$ -rule. The second problem is that infinite reduction sequences occur with ς -rule: For example, $\langle x \rangle \text{inl} \bullet \beta$ is reduced to $\langle (x \bullet \alpha). \alpha \rangle \text{inl} \bullet \beta$ by $(\eta R)_v^+$. Since $(x \bullet \alpha). \alpha$ is not a value,

it can be reduced to $((x \bullet \alpha). \alpha \bullet y. (\langle y \rangle \text{inl} \bullet \gamma)). \gamma \bullet \beta$ by $(\zeta \vee_1)_v$. Then, we have $\langle x \rangle \text{inl} \bullet \beta$ again by $(\beta L)_v$ and $(\beta R)_v$ -rules. Tzevelekos [26] assumed additional conditions on η -expansion rules, and showed strong normalization and the Church-Rosser properties of the call-by-value and call-by-name DC under his conditions. However, his approach does not solve the first problem. One simple solution for the both problems is to replace η -expansion by η -reduction. For this reason, we will adopt η -reduction in our call-by-value and call-by-name systems.

The dual calculus considered in [28] has η -rules for conjunction, disjunction, and negation. These rules could be defined naturally because the system in [28] was based on equations. However, we cannot define these η -rules naively in the call-by-value and call-by-name reduction systems of DC since these rules break the Church-Rosser property: The call-by-value $(\eta \vee)$ -rule defined in [28] is $[x. (\langle x \rangle \text{inl} \bullet K), y. (\langle y \rangle \text{inr} \bullet K)] = K$, where K has type $A \vee B$. Suppose that we add $(\eta \vee)$ -reduction rule $[x_1. (\langle x_1 \rangle \text{inl} \bullet K), x_2. (\langle x_2 \rangle \text{inr} \bullet K)] \rightarrow_{\text{DC}}^v K$ to the call-by-value DC. Then the statement $[x_1. (\langle x_1 \rangle \text{inl} \bullet y. (z \bullet \alpha)), x_2. (\langle x_2 \rangle \text{inr} \bullet y. (z \bullet \alpha))]$ has two normal forms $[x_1. (z \bullet \alpha), x_2. (z \bullet \alpha)]$ and $y. (z \bullet \alpha)$. Suppose that we add $(\eta \vee)$ -expansion rule $K \rightarrow_{\text{DC}}^v [x_1. (\langle x_1 \rangle \text{inl} \bullet K), x_2. (\langle x_2 \rangle \text{inr} \bullet K)]$ to the call-by-value DC. The statement $x \bullet y. (z \bullet \alpha)$ (the variable z and the covariable α have type X , and the variables x and y have type $A \vee B$) is reduced to $z \bullet \alpha$ by (βL) -rule. The statement $x \bullet y. (z \bullet \alpha)$ is also expanded to $x \bullet [x_1. (\langle x_1 \rangle \text{inl} \bullet y. (z \bullet \alpha)), x_2. (\langle x_2 \rangle \text{inr} \bullet y. (z \bullet \alpha))]$ by $(\eta \vee)$ -rule, and then it is reduced to $x \bullet [x_1. (z \bullet \alpha), x_2. (z \bullet \alpha)]$ by $(\beta L)_v$ -rule. These two results are never confluent since the first one $z \bullet \alpha$ cannot produce a cotermin of the form $[K, L]$, and the bracket $[..]$ in the second one $x \bullet [x_1. (z \bullet \alpha), x_2. (z \bullet \alpha)]$ cannot be eliminated.

The call-by-name reduction relation of DC is defined as follows.

Definition 7.3 (Call-by-name reduction rules of DC). The call-by-name reduction relation $\rightarrow_{\text{DC}}^n$ of DC is defined from the following rules.

$$\begin{aligned}
(\beta \wedge_1)_n & \langle M, N \rangle \bullet \text{fst}[P] \rightarrow_{\text{DC}}^n M \bullet P, \\
(\beta \wedge_2)_n & \langle M, N \rangle \bullet \text{snd}[P] \rightarrow_{\text{DC}}^n N \bullet P, \\
(\beta \vee_1)_n & \langle M \rangle \text{inl} \bullet [P, Q] \rightarrow_{\text{DC}}^n M \bullet P, \\
(\beta \vee_2)_n & \langle M \rangle \text{inr} \bullet [P, Q] \rightarrow_{\text{DC}}^n M \bullet Q, \\
(\beta \neg)_n & [K] \text{not} \bullet \text{not} \langle M \rangle \rightarrow_{\text{DC}}^n M \bullet K, \\
(\beta R)_n & (S). \alpha \bullet P \rightarrow_{\text{DC}}^n S[P/\alpha], \\
(\beta L)_n & M \bullet x.(S) \rightarrow_{\text{DC}}^n S[M/x], \\
(\zeta \wedge_1)_n & \text{fst}[\mathcal{K}] \rightarrow_{\text{DC}}^n x.((x \bullet \text{fst}[\alpha]). \alpha \bullet \mathcal{K}), \\
(\zeta \wedge_2)_n & \text{snd}[\mathcal{K}] \rightarrow_{\text{DC}}^n x.((x \bullet \text{snd}[\alpha]). \alpha \bullet \mathcal{K}), \\
(\zeta \vee_1)_n & [\mathcal{K}, L] \rightarrow_{\text{DC}}^n x.((x \bullet [\alpha, L]). \alpha \bullet \mathcal{K}), \\
(\zeta \vee_2)_n & [P, \mathcal{K}] \rightarrow_{\text{DC}}^n x.((x \bullet [P, \alpha]). \alpha \bullet \mathcal{K}), \\
(\eta R)_n^+ & M \rightarrow_{\text{DC}}^n (M \bullet \alpha). \alpha, \quad \text{and} \\
(\eta L)_n^+ & K \rightarrow_{\text{DC}}^n x.(x \bullet K),
\end{aligned}$$

where \mathcal{K} is not a covalue, and x and α in $(\zeta \wedge_1)_n, (\zeta \wedge_2)_n, (\zeta \vee_1)_n, (\zeta \vee_2)_n, (\eta L)_n^+$ and $(\eta R)_n^+$ are fresh.

This system is obtained from the call-by-name dual calculus given in [27] by removing the implication.

As mentioned in [27], an implication $A \supset B$ can be defined as $\neg A \vee B$ under call-by-name. Hence, covalues for implication, and a call-by-name reduction rules for implication given in the original system can be replaced in terms of other connectives.

The call-by-value reduction and the call-by-name reduction are dual strategies in DC.

Proposition 7.4 (Duality between call-by-value and call-by-name in DC [27]). *Let D and E be expressions of DC. Then, $D \rightarrow_{\text{DC}}^v E$ iff $(D)^\circ \rightarrow_{\text{DC}}^n (E)^\circ$, where $(-)^{\circ}$ is the duality transformation defined in the section 2.*

Now we will introduce the call-by-value and call-by-name variants of $\text{DC}\mu\nu$. We first consider a call-by-value restriction of $\text{DC}\mu\nu$ (called *weak call-by-value* $\text{DC}\mu\nu$) which is given by simply restricting the reduction rules of $\text{DC}\mu\nu$. This restricted system satisfies both strong normalization and the Church-Rosser properties. However, this system is rather weak since it lacks the ζ -rules. The call-by-value $\text{DC}\mu\nu$ (denoted by $\text{CBV DC}\mu\nu$) is obtained by adding the ζ -rules to the weak call-by-value $\text{DC}\mu\nu$. The weak call-by-name $\text{DC}\mu\nu$ and the call-by-name $\text{DC}\mu\nu$ (denoted by $\text{CBN DC}\mu\nu$) are also considered. The call-by-name $\text{DC}\mu\nu$ is the dual system of the call-by-value $\text{DC}\mu\nu$.

We first define the notion of values and covalues in $\text{DC}\mu\nu$.

Definition 7.5 (Values and covalues of $\text{DC}\mu\nu$). The values (denoted by V, W, \dots) and the covalues (denoted by P, Q, \dots) of $\text{DC}\mu\nu$ are defined by the following grammar:

$$\begin{aligned} V &::= x \mid \langle V, V \rangle \mid \langle V \rangle \text{inl} \mid \langle V \rangle \text{inr} \mid [K] \text{not} \mid \text{in}^{\mu X.A} \langle V \rangle \mid \text{coitr}_x^A \langle M, V \rangle, \\ P &::= \alpha \mid [P, P] \mid \text{fst}[P] \mid \text{snd}[P] \mid \text{not} \langle M \rangle \mid \text{out}^{\nu X.A} [P] \mid \text{itr}_\alpha^A [K, P], \end{aligned}$$

where M is a term and K is a cotermin of $\text{DC}\mu\nu$.

The set of values of $\text{DC}\mu\nu$ is a subset of terms of $\text{DC}\mu\nu$. The set of covalues of $\text{DC}\mu\nu$ is a subset of cotermins of $\text{DC}\mu\nu$. Note that the above definition is a straightforward extension of the definition of values and covalues in DC.

The set of values and covalues are closed under substitution of values and covalues, respectively.

Lemma 7.6. *Let V and W be values, and P and Q be covalues of $\text{DC}\mu\nu$. The following claims hold.*

- (1) $V[W/x]$ is a value of $\text{DC}\mu\nu$.
- (2) $P[Q/\alpha]$ is a covalue of $\text{DC}\mu\nu$.

Proof. They are straightforwardly proved by induction on V and P . □

The types, expressions, and typing rules of the weak call-by-value and the weak call-by-name $\text{DC}\mu\nu$ are the same as them of $\text{DC}\mu\nu$. The reduction relation of the weak call-by-value $\text{DC}\mu\nu$ is given as follows.

Definition 7.7 (Reduction rules of the weak call-by-value $\text{DC}\mu\nu$). The reduction relation $\rightarrow_{\text{wCBV}}$ of the weak call-by-value $\text{DC}\mu\nu$ is defined as the compatible closure of the reduction rules $(\beta\wedge_1)_v$, $(\beta\wedge_2)_v$, $(\beta\vee_1)_v$, $(\beta\vee_2)_v$, $(\beta\neg)_v$, $(\beta R)_v$, $(\beta L)_v$, and the following reduction rules:

$$\begin{aligned} (\beta\mu)_v & \text{in}^{\mu X.C} \langle V \rangle \bullet \text{itr}_\alpha^A [K, L] \rightarrow_{\text{wCBV}} (V \bullet \text{mono}_{\mu X.C, A, \beta}^{X.C} \{ \text{itr}_\alpha^A [K, \beta], K \}) \bullet \alpha \bullet L, \\ (\beta\nu)_v & \text{coitr}_x^A \langle M, V \rangle \bullet \text{out}^{\nu X.C} [K] \rightarrow_{\text{wCBV}} V \bullet x.(\text{mono}_{A, \nu X.C, z}^{X.C} \{ \text{coitr}_x^A \langle M, z \rangle, M \} \bullet K), \\ (\eta R)_v & (M \bullet \alpha) \bullet \alpha \rightarrow_{\text{wCBV}} M, \\ (\eta L)_v & x.(x \bullet K) \rightarrow_{\text{wCBV}} K, \end{aligned}$$

where x and α are fresh in $(\eta L)_v$ and $(\eta R)_v$, respectively.

The reduction relation of the weak call-by-name $\text{DC}\mu\nu$ is given as follows.

Definition 7.8 (Reduction rules of the weak call-by-name $\text{DC}\mu\nu$). The reduction relation $\rightarrow_{\text{wCBN}}$ of the weak call-by-name $\text{DC}\mu\nu$ is defined as the compatible closure of the reduction rules $(\beta\wedge_1)_n$, $(\beta\wedge_2)_n$, $(\beta\vee_1)_n$, $(\beta\vee_2)_n$, $(\beta\neg)_n$, $(\beta R)_n$, $(\beta L)_n$, and the following reduction rules:

$$\begin{aligned}
(\beta\mu)_n \quad & \text{in}^{\mu X.C} \langle M \rangle \bullet \text{itr}_\alpha^A[K, P] \longrightarrow_{\text{wCBN}} (M \bullet \text{mono}_{\mu X.C, A, \beta}^{X.C} \{ \text{itr}_\alpha^A[K, \beta], K \}) . \alpha \bullet P, \\
(\beta\nu)_n \quad & \text{coitr}_x^A \langle M, N \rangle \bullet \text{out}^{\nu X.C}[P] \longrightarrow_{\text{wCBN}} M \bullet x . (\text{mono}_{A, \nu X.C, z}^{X.C} \{ \text{coitr}_x^A \langle M, z \rangle, M \} \bullet P), \\
(\eta R)_n \quad & (M \bullet \alpha) . \alpha \longrightarrow_{\text{wCBN}} M, \\
(\eta L)_n \quad & x . (x \bullet K) \longrightarrow_{\text{wCBN}} K,
\end{aligned}$$

where x and α are fresh in $(\eta L)_n$ and $(\eta R)_n$, respectively.

The weak call-by-value reduction and the weak call-by-name reduction are dual strategies.

Proposition 7.9 (Duality between weak call-by-value and weak call-by-name in $\text{DC}\mu\nu$). *Let D and E be expressions of $\text{DC}\mu\nu$. Then, $D \longrightarrow_{\text{wCBV}} E$ iff $(D)^\circ \longrightarrow_{\text{wCBN}} (E)^\circ$, where $(-)^^\circ$ is the duality transformation of $\text{DC}\mu\nu$ defined in the section 3.*

The rules $(\beta\nu)_v$, $(\beta R)_v$, $(\eta L)_v$, and $(\eta R)_v$ are the same as $(\beta\nu)$, (βR) , (ηL) , and (ηR) -rules of $\text{DC}\mu\nu$, respectively. The rules $(\beta\wedge_1)_v$, $(\beta\wedge_2)_v$, $(\beta\vee_1)_v$, $(\beta\vee_2)_v$, $(\beta L)_v$, $(\beta\mu)_v$, and $(\beta\nu)_v$ are just restrictions of the rules $(\beta\wedge_1)$, $(\beta\wedge_2)$, $(\beta\vee_1)$, $(\beta\vee_2)$, (βL) , $(\beta\mu)$, and $(\beta\nu)$, respectively. The situation of the call-by-name case is similar to the call-by-value case. Hence, we can easily obtain the following proposition.

Proposition 7.10. *Let D and E be expressions in $\text{DC}\mu\nu$. Then the following claims hold.*

- (1) *If $D \longrightarrow_{\text{wCBV}} E$, then $D \longrightarrow_{\text{DC}\mu\nu} E$.*
- (2) *If $D \longrightarrow_{\text{wCBN}} E$, then $D \longrightarrow_{\text{DC}\mu\nu} E$.*

From the above proposition and the strong normalization result of $\text{DC}\mu\nu$ (Theorem 6.8), we have the strong normalization of the weak call-by-value and the weak call-by-name reduction relations.

Proposition 7.11 (Strong normalization of the weak CBV and CBN $\text{DC}\mu\nu$). *We have the following.*

- (1) *Every typable expression is strongly normalizing in the weak call-by-value $\text{DC}\mu\nu$.*
- (2) *Every typable expression is strongly normalizing in the weak call-by-name $\text{DC}\mu\nu$.*

The reduction relations $\longrightarrow_{\text{wCBV}}$ and $\longrightarrow_{\text{wCBN}}$ of $\text{DC}\mu\nu$ satisfy the Church-Rosser property. We first recall the definition of the Church-Rosser property.

Definition 7.12 (Church-Rosser property). Let A be a set and \rightarrow be a reduction relation on A . We write $b \leftarrow a \rightarrow c$ if both $a \rightarrow b$ and $a \rightarrow c$ hold. We also write $b \rightarrow a \leftarrow c$ if both $b \rightarrow a$ and $c \rightarrow a$ hold.

- (1) The reduction relation \rightarrow satisfies the diamond property if, for all $a, b, c \in A$, the relation $b \leftarrow a \rightarrow c$ implies that there exists $d \in A$ such that $b \rightarrow d \leftarrow c$.
- (2) The reduction relation \rightarrow satisfies the Church-Rosser property if \rightarrow^* satisfies the diamond property, where \rightarrow^* is the reflexive transitive closure of \rightarrow .

From now on, we concentrate to show the Church-Rosser property of $\longrightarrow_{\text{wCBV}}$. The Church-Rosser property of $\longrightarrow_{\text{wCBN}}$ can be obtained from the result of $\longrightarrow_{\text{wCBV}}$ and the duality (Proposition 7.9). In order to show the Church-Rosser property of $\longrightarrow_{\text{wCBV}}$, we will use the parallel reduction technique. The definition of the parallel reduction relation is given as follows.

Definition 7.13 (Parallel reduction of the weak call-by-value $\text{DC}\mu\nu$). The parallel reduction relation (denoted by \Rightarrow) of the weak call-by-value $\text{DC}\mu\nu$ is defined inductively from the following rules.

$$\begin{aligned}
& x \Rightarrow x \text{ and } \alpha \Rightarrow \alpha \text{ for any variable } x \text{ and covariable } \alpha. \\
& \langle M, N \rangle \Rightarrow \langle M', N' \rangle \quad \text{if } M \Rightarrow M' \text{ and } N \Rightarrow N'. \\
& [K, L] \Rightarrow [K', L'] \quad \text{if } K \Rightarrow K' \text{ and } L \Rightarrow L'. \\
& \langle M \rangle \text{inl} \Rightarrow \langle M' \rangle \text{inl}, \langle M \rangle \text{inr} \Rightarrow \langle M' \rangle \text{inr}, \text{ and } \text{not} \langle M \rangle \Rightarrow \text{not} \langle M' \rangle \quad \text{if } M \Rightarrow M'.
\end{aligned}$$

$\text{fst}[K] \Rightarrow \text{fst}[K'], \text{snd}[K] \Rightarrow \text{snd}[K'], \text{ and } [K]\text{not} \Rightarrow [K']\text{not} \quad \text{if } K \Rightarrow K'.$
 $\text{in}^{\mu X.A}\langle M \rangle \Rightarrow \text{in}^{\mu X.A}\langle M' \rangle \quad \text{if } M \Rightarrow M'.$
 $\text{out}^{\nu X.A}[K] \Rightarrow \text{out}^{\nu X.A}[K'] \quad \text{if } K \Rightarrow K'.$
 $\text{coitr}_x^A\langle M, N \rangle \Rightarrow \text{coitr}_x^A\langle M', N' \rangle \quad \text{if } M \Rightarrow M' \text{ and } N \Rightarrow N'.$
 $\text{itr}_\alpha^A[K, L] \Rightarrow \text{itr}_\alpha^A[K', L'] \quad \text{if } K \Rightarrow K' \text{ and } L \Rightarrow L'.$
 $M \bullet K \Rightarrow M' \bullet K' \quad \text{if } M \Rightarrow M' \text{ and } K \Rightarrow K'.$
 $(S).\alpha \Rightarrow (S').\alpha \text{ and } x.(S) \Rightarrow x.(S') \quad \text{if } S \Rightarrow S'.$
 $M \bullet x.(S) \Rightarrow S'[V/x] \quad \text{if } M \Rightarrow V \text{ and } S \Rightarrow S'.$
 $(S).\alpha \bullet K \Rightarrow S'[K'/\alpha] \quad \text{if } K \Rightarrow K' \text{ and } S \Rightarrow S'.$
 $\langle M, N \rangle \bullet \text{fst}[K] \Rightarrow V \bullet K' \quad \text{if } M \Rightarrow V, N \Rightarrow W, \text{ and } K \Rightarrow K'.$
 $\langle M, N \rangle \bullet \text{snd}[K] \Rightarrow W \bullet K' \quad \text{if } M \Rightarrow V, N \Rightarrow W, \text{ and } K \Rightarrow K'.$
 $\langle M \rangle \text{inl} \bullet [K, L] \Rightarrow V \bullet K' \quad \text{if } M \Rightarrow V, K \Rightarrow K'.$
 $\langle M \rangle \text{inl} \bullet [K, L] \Rightarrow V \bullet L' \quad \text{if } M \Rightarrow V, L \Rightarrow L'.$
 $[K]\text{not} \bullet \text{not}\langle M \rangle \Rightarrow M' \bullet K' \quad \text{if } M \Rightarrow M' \text{ and } K \Rightarrow K'.$
 $\text{in}^{\mu X.C}\langle M \rangle \bullet \text{itr}_\alpha^A[K, L] \Rightarrow (V \bullet \text{mono}_{\mu X.C, A, \beta}^{X.C}\{\text{itr}_\alpha^A[K', \beta], K'\}).\alpha \bullet L' \quad \text{if } M \Rightarrow V, K \Rightarrow K',$
 $\text{ and } L \Rightarrow L'.$
 $\text{coitr}_x^A\langle M, N \rangle \bullet \text{out}^{\nu X.C}[K] \Rightarrow V \bullet x.(\text{mono}_{A, \nu X.C, z}^{X.C}\{\text{coitr}_x^A\langle M', z \rangle, M'\} \bullet K') \quad \text{if } M \Rightarrow M',$
 $N \Rightarrow V, \text{ and } K \Rightarrow K'.$
 $(M \bullet \alpha).\alpha \Rightarrow M' \quad \text{if } M \Rightarrow M' \text{ and } \alpha \text{ is not free in } M.$
 $x.(x \bullet K) \Rightarrow K' \quad \text{if } K \Rightarrow K' \text{ and } x \text{ is not free in } K.$

The parallel reduction of the weak call-by-value $\text{DC}\mu\nu$ satisfies the following basic properties.

Lemma 7.14. *Let M be a term, V and V' be values, K and K' be coterms, and D and D' be expressions of $\text{DC}\mu\nu$. Then the following hold.*

- (1) *Suppose $D \Rightarrow E$. If D is a term, then E is also a term. If D is a cotermin, then E is also a cotermin. If D is a statement, then E is also a statement. If D is a value, then E is also a value.*
- (2) $D \Rightarrow D$.
- (3) *If $M \Rightarrow V$ and $D \Rightarrow D'$, then $D[M/x] \Rightarrow D'[V/x]$.*
- (4) *If $K \Rightarrow K'$ and $D \Rightarrow D'$, then $D[K/\alpha] \Rightarrow D'[K'/\alpha]$.*

Proof. The claim (1) is shown by induction on the definition of \Rightarrow . The claim (2) is shown by induction on D .

The claim (3) is shown by induction on $D \Rightarrow D'$ with Lemma 7.6. We show the case that $N_0 \bullet y.(T_0) \Rightarrow T_1[W/y]$ is derived from $N_0 \Rightarrow W$ and $T_0 \Rightarrow T_1$. By the induction hypothesis, we have $N_0[M/x] \Rightarrow W[V/x]$ and $T_0[M/x] \Rightarrow T_1[V/x]$. By Lemma 7.6, $W[V/x]$ is a value. Hence we have $(N_0 \bullet y.(T_0))[M/x] = (N_0[M/x]) \bullet y.(T_0[M/x]) \Rightarrow T_1[V/x][W[V/x]/y] = T_1[W/y][V/x]$. The other cases are straightforwardly proved by the induction hypothesis.

The claim (4) is shown by induction on $D \Rightarrow D'$. □

Lemma 7.15. *Let D and D' be expressions of $\text{DC}\mu\nu$. Then the following claims hold.*

- (1) *If $D \longrightarrow_{\text{wCBV}} E$, then $D \Rightarrow E$.*
- (2) *If $D \Rightarrow E$, then $D \longrightarrow_{\text{wCBV}}^* E$.*
- (3) *The parallel reduction relation \Rightarrow satisfies the diamond property, that is, if the relation $D_1 \Leftarrow D \Rightarrow D_2$ holds, then there exists E such that $D_1 \Rightarrow E \Leftarrow D_2$.*

Proof. The claim (1) is shown by induction on the definition of $\longrightarrow_{\text{wCBV}}$. The claim (2) is shown by induction on the definition of \Rightarrow .

The claim (3) is shown by induction on D . We show the case that D is the shape of $(S).\alpha \bullet x.(T)$, D_1 is $S'[L/\alpha]$, and D_2 is $T'[V/x]$ with the conditions $S \Rightarrow S'$, $T \Rightarrow T'$, $x.T \Rightarrow L$, and $(S).\alpha \Rightarrow V$. Recall that a critical pair in $\text{DC}\mu\nu$ occurs in this shape. This case is most important to see that this critical pair is avoided in the weak call-by-value $\text{DC}\mu\nu$.

From the definition of the parallel reduction and $(S).\alpha \Rightarrow V$, we have $S = M \bullet \alpha$, $M \Rightarrow V$, and α is not free in M . Then, from $M \bullet \alpha = S \Rightarrow S'$, we have the following two cases: (i) $S' = M' \bullet \alpha$ and $M \Rightarrow M'$ for some M' , or (ii) $M = (S_0).\beta$, $S' = S_0'[\alpha/\beta]$, and $S_0 \Rightarrow S_0'$ for some S_0 and S_0' . From the condition $x.(T) \Rightarrow L$, we also have the following two cases: (a) $T = x \bullet K$, x is not free in K , and $K \Rightarrow L$ for some K and L , or (b) $L = x.(T'')$ and $T \Rightarrow T''$ for some T'' .

The case of (i). We have $D_1 = (S')[L/\alpha] = (M' \bullet \alpha)[L/\alpha] = M' \bullet L$. By the induction hypothesis and $V \Leftarrow M \Rightarrow M'$, there exists a term W such that $V \Rightarrow W \Leftarrow M'$. From Lemma 7.14 (1), W is a value. We then consider the subcases (a) and (b).

The subcase of (a). From the condition $K \Rightarrow L$ and $T = x \bullet K$, we have $(x \bullet L) \Leftarrow T \Rightarrow T'$. By the induction hypothesis, there exists a statement \tilde{T} such that $(x \bullet L) \Rightarrow \tilde{T} \Leftarrow T'$. Hence we have $D_1 = (x \bullet L)[M'/x] \Rightarrow \tilde{T}[W/x] \Leftarrow T'[V/x] = D_2$ from $M' \Rightarrow W \Leftarrow V$ and Lemma 7.14 (3).

The subcase of (b). By the induction hypothesis and $T' \Leftarrow T \Rightarrow T''$, there exists \tilde{T} such that $T' \Rightarrow \tilde{T} \Leftarrow T''$. Hence we have $D_2 = T'[V/x] \Rightarrow \tilde{T}[W/x]$ by Lemma 7.14 (3) and $V \Rightarrow W$. We also have $D_1 = M' \bullet L = M' \bullet x.(T'') \Rightarrow \tilde{T}[W/x]$ from $M' \Rightarrow W$ and $T'' \Rightarrow \tilde{T}$. Therefore $D_1 \Rightarrow \tilde{T}[W/x] \Leftarrow D_2$ holds.

The case of (ii). We first claim that, for any S and V , if $(S).\alpha \Rightarrow V$, then there is some M such that $S = M \bullet \alpha$, $M \Rightarrow V$, and α is not free in M . This claim is easily obtained from the definition of the parallel reduction. In this case, we have $V \Leftarrow M = (S_0).\beta \Rightarrow (S_0').\beta$. By the induction hypothesis and Lemma 7.14 (1), there is a value W such that $V \Rightarrow W \Leftarrow (S_0').\beta$. Then, there exists a N such that $S_0' = N \bullet \beta$, $N \Rightarrow W$, and β is not free in N from the above claim. Hence we have $D_1 = S'[L/\alpha] = S_0'[\alpha/\beta][L/\alpha] = S_0'[L/\beta] = (N \bullet \beta)[L/\beta] = N \bullet L$. We then consider the subcases (a) and (b).

The subcase of (a). By the induction hypothesis, there exists a statement \tilde{T} such that $(x \bullet L) \Rightarrow \tilde{T} \Leftarrow T'$. Hence we have $D_1 = (x \bullet L)[N/x] \Rightarrow \tilde{T}[W/x] \Leftarrow T'[V/x] = D_2$ from $N \Rightarrow W \Leftarrow V$ and Lemma 7.14 (3).

The subcase of (b). By the induction hypothesis and $T' \Leftarrow T \Rightarrow T''$, there exists \tilde{T} such that $T' \Rightarrow \tilde{T} \Leftarrow T''$. Hence we have $D_2 = T'[V/x] \Rightarrow \tilde{T}[W/x]$ by Lemma 7.14 (3) and $V \Rightarrow W$. We also have $D_1 = N \bullet L = N \bullet x.(T'') \Rightarrow \tilde{T}[W/x]$ from $N \Rightarrow W$ and $T'' \Rightarrow \tilde{T}$. Therefore $D_1 \Rightarrow \tilde{T}[W/x] \Leftarrow D_2$ holds.

The other cases are also proved by the induction hypothesis. \square

From Lemma 7.15, we can obtain the Church-Rosser property of the weak call-by-value $\text{DC}\mu\nu$.

Proposition 7.16. *The reduction relations $\longrightarrow_{\text{wCBV}}$ and $\longrightarrow_{\text{wCBN}}$ of $\text{DC}\mu\nu_{\text{cbv}}$ satisfy the Church-Rosser property.*

Proof. We first show the Church-Rosser property of $\longrightarrow_{\text{wCBV}}$. Suppose that $D \xrightarrow{*}_{\text{wCBV}} D'$ and $D \xrightarrow{*}_{\text{wCBV}} D''$ hold. We will show that there exists some E such that $D' \xrightarrow{*}_{\text{wCBV}} E$ and $D'' \xrightarrow{*}_{\text{wCBV}} E$. We have $D = D_{00} \xrightarrow{\text{wCBV}} D_{01} \xrightarrow{\text{wCBV}} \dots \xrightarrow{\text{wCBV}} D_{0n} = D'$ and $D \xrightarrow{\text{wCBV}} D_{10} \xrightarrow{\text{wCBV}} \dots \xrightarrow{\text{wCBV}} D_{1m} = D''$ for some $n, m \geq 0$. By Lemma 7.15 (1), $D \Rightarrow D_{01} \Rightarrow \dots \Rightarrow D_{0n} = D'$ and $D \Rightarrow D_{10} \Rightarrow \dots \Rightarrow D_{1m} = D''$ hold. By the diamond property of \Rightarrow , there exists $D_{(i+1)(j+1)}$ such that $D_{i(j+1)} \Rightarrow D_{(i+1)(j+1)} \Leftarrow D_{(i+1)j}$ for each $0 \leq i \leq n-1$ and $0 \leq j \leq m-1$. Hence we have $D' = D_{0n} \Rightarrow D_{1n} \Rightarrow \dots \Rightarrow D_{mn}$ and $D'' = D_{m0} \Rightarrow D_{m1} \Rightarrow \dots \Rightarrow D_{mn}$. By Lemma 7.15 (2), we can replace \Rightarrow by $\xrightarrow{*}_{\text{wCBV}}$. Therefore, we have $D' \xrightarrow{*}_{\text{wCBV}} D_{mn}$ and $D'' \xrightarrow{*}_{\text{wCBV}} D_{mn}$.

The Church-Rosser property of $\longrightarrow_{\text{wCBN}}$ is shown by the former result and the duality between $\longrightarrow_{\text{wCBV}}$ and $\longrightarrow_{\text{wCBN}}$ (Proposition 7.9). \square

We will next define the call-by-value and the call-by-name $\text{DC}\mu\nu$, which we call $\text{CBV DC}\mu\nu$ and $\text{CBN DC}\mu\nu$. The types, expressions, and typing rules of $\text{CBV DC}\mu\nu$ and $\text{CBN DC}\mu\nu$ are the same as them of $\text{DC}\mu\nu$.

The reduction relation of $\text{CBV DC}\mu\nu$ is obtained by adding ς -rules to the weak call-by-value $\text{DC}\mu\nu$.

Definition 7.17 (Reduction relation of $\text{CBV DC}\mu\nu$). The reduction relation $\longrightarrow_{\text{CBV}}$ of $\text{CBV DC}\mu\nu$ is defined by the compatible closure of the reduction rules of the weak call-by-value $\text{DC}\mu\nu$ and $(\varsigma\wedge_1)_v$, $(\varsigma\wedge_2)_v$, $(\varsigma\vee_1)_v$, $(\varsigma\vee_2)_v$, and the following reduction rules:

$$\begin{aligned} (\varsigma\mu)_v & \text{ in}^{\mu X.C} \langle M \rangle \longrightarrow_{\text{CBV}} (M \bullet x.(\text{in}^{\mu X.C} \langle x \rangle \bullet \alpha)).\alpha, \\ (\varsigma\nu)_v & \text{ coitr}_y^A \langle M, M \rangle \longrightarrow_{\text{CBV}} (M \bullet x.(\text{coitr}_y^A \langle M, x \rangle \bullet \alpha)).\alpha, \end{aligned}$$

where M is not a value of $\text{DC}\mu\nu$, and the variable x and the covariable α in $(\varsigma\mu)_v$, $(\varsigma\nu)_v$ are fresh.

We sometimes write $(\beta)_v$ to mean $(\beta\wedge_1)_v$, $(\beta\wedge_2)_v$, $(\beta\vee_1)_v$, $(\beta\vee_2)_v$, $(\beta\neg)_v$, $(\beta\mu)_v$, $(\beta\nu)_v$, $(\beta L)_v$, or $(\beta R)_v$ -rule. We write $(\eta)_v$ to mean $(\eta L)_v$ or $(\eta R)_v$ -rule. We also write $(\varsigma)_v$ to mean $(\varsigma\wedge_1)_v$, $(\varsigma\wedge_2)_v$, $(\varsigma\vee_1)_v$, $(\varsigma\vee_2)_v$, $(\varsigma\mu)_v$, or $(\varsigma\nu)_v$ -rule.

The reduction relation of $\text{CBN DC}\mu\nu$ is obtained by adding ς -rules to the weak call-by-name $\text{DC}\mu\nu$.

Definition 7.18 (Reduction relation of $\text{CBN DC}\mu\nu$). The reduction relation $\longrightarrow_{\text{CBN}}$ of $\text{CBN DC}\mu\nu$ is defined by the compatible closure of the reduction rules of the weak call-by-name $\text{DC}\mu\nu$ and $(\varsigma\wedge_1)_n$, $(\varsigma\wedge_2)_n$, $(\varsigma\vee_1)_n$, $(\varsigma\vee_2)_n$, and the following reduction rules:

$$\begin{aligned} (\varsigma\mu)_n & \text{ itr}_\beta^A [K, \mathcal{K}] \longrightarrow_{\text{CBN}} x.((x \bullet \text{itr}_\beta^A [K, \alpha]).\alpha \bullet \mathcal{K}), \\ (\varsigma\nu)_n & \text{ out}^{\nu X.C} [\mathcal{K}] \longrightarrow_{\text{CBN}} x.((x \bullet \text{out}^{\nu X.C} [\alpha]).\alpha \bullet \mathcal{K}), \end{aligned}$$

where \mathcal{K} is not a covalue of $\text{DC}\mu\nu$, and the variable x and the covariable α in $(\varsigma\mu)_n$, $(\varsigma\nu)_n$ are fresh.

From the above definitions, $\text{CBV DC}\mu\nu$ includes the weak call-by-value $\text{DC}\mu\nu$, and $\text{CBN DC}\mu\nu$ includes the weak call-by-name $\text{DC}\mu\nu$. That is, the following lemma holds.

Lemma 7.19. *Let D and E be expressions of $\text{DC}\mu\nu$. Then, the following claims hold.*

- (1) *If $D \longrightarrow_{\text{wCBV}} E$, then $D \longrightarrow_{\text{CBV}} E$.*
- (2) *If $D \longrightarrow_{\text{wCBN}} E$, then $D \longrightarrow_{\text{CBN}} E$.*

The call-by-value $\text{DC}\mu\nu$ is dual to the call-by-name $\text{DC}\mu\nu$.

Proposition 7.20 (Duality between call-by-value and call-by-name in $\text{DC}\mu\nu$). *Let D and E be expressions of $\text{DC}\mu\nu$. Then, $D \longrightarrow_{\text{CBV}} E$ iff $(D)^\circ \longrightarrow_{\text{CBN}} (E)^\circ$, where $(-)^{\circ}$ is the duality transformation of $\text{DC}\mu\nu$ defined in the section 3.*

The call-by-value and call-by-name $\text{DC}\mu\nu$ satisfy both the Church-Rosser and strong normalization properties. We will concentrate to show these properties of $\text{CBV DC}\mu\nu$. The proof will be performed by giving a transformation from $\text{CBV DC}\mu\nu$ into the weak call-by-value $\text{DC}\mu\nu$. The transformation $(-)^{\circ}$ given as follows.

Definition 7.21. Let D be an expression of $\text{DC}\mu\nu$. The expression $(D)^\circ$ of $\text{DC}\mu\nu$ is defined inductively as follows.

$$\begin{aligned}
(x)^\circ &= x, \\
(\langle V, W \rangle)^\circ &= \langle (V)^\circ, (W)^\circ \rangle, \\
(\langle V, \mathcal{N} \rangle)^\circ &= ((\mathcal{N})^\circ \bullet y.(\langle (V)^\circ, y \rangle \bullet \alpha)).\alpha, \\
(\langle \mathcal{M}, W \rangle)^\circ &= ((\mathcal{M})^\circ \bullet x.(\langle x, (W)^\circ \rangle \bullet \alpha)).\alpha, \\
(\langle \mathcal{M}, \mathcal{N} \rangle)^\circ &= ((\mathcal{M})^\circ \bullet x.((\mathcal{N})^\circ \bullet y.(\langle x, y \rangle \bullet \beta)).\beta \bullet \alpha).\alpha, \\
(\langle V \rangle \text{inl})^\circ &= \langle (V)^\circ \rangle \text{inl}, \\
(\langle \mathcal{M} \rangle \text{inl})^\circ &= ((\mathcal{M})^\circ \bullet x.(\langle x \rangle \text{inl} \bullet \alpha)).\alpha, \\
(\langle V \rangle \text{inr})^\circ &= \langle (V)^\circ \rangle \text{inr}, \\
(\langle \mathcal{M} \rangle \text{inr})^\circ &= ((\mathcal{M})^\circ \bullet x.(\langle x \rangle \text{inr} \bullet \alpha)).\alpha, \\
([K] \text{not})^\circ &= [(K)^\circ] \text{not}, \\
(\text{in}^{\mu X.A} \langle \mathcal{M} \rangle)^\circ &= ((\mathcal{M})^\circ \bullet x.(\text{in}^{\mu X.A} \langle x \rangle \bullet \alpha)).\alpha, \\
(\text{in}^{\mu X.A} \langle V \rangle)^\circ &= \text{in}^{\mu X.A} \langle (V)^\circ \rangle, \\
(\text{coitr}_z^A \langle \mathcal{M}, V \rangle)^\circ &= \text{coitr}_z^A \langle (\mathcal{M})^\circ, (V)^\circ \rangle, \\
(\text{coitr}_z^A \langle \mathcal{M}, \mathcal{N} \rangle)^\circ &= ((\mathcal{N})^\circ \bullet y.(\text{coitr}_z^A \langle (\mathcal{M})^\circ, y \rangle \bullet \alpha)).\alpha, \\
((S).\alpha)^\circ &= ((S)^\circ).\alpha, \\
(\alpha)^\circ &= \alpha, \\
([K, L])^\circ &= [(K)^\circ, (L)^\circ], \\
(\text{fst}[K])^\circ &= \text{fst}[(K)^\circ], \\
(\text{snd}[K])^\circ &= \text{snd}[(K)^\circ], \\
(\text{not} \langle \mathcal{M} \rangle)^\circ &= \text{not} \langle (\mathcal{M})^\circ \rangle, \\
(\text{out}^{\nu X.A} [K])^\circ &= \text{out}^{\nu X.A} [(K)^\circ], \\
(\text{itr}_\gamma^A [K, L])^\circ &= \text{itr}_\gamma^A [(K)^\circ, (L)^\circ], \\
(x.(S))^\circ &= x.((S)^\circ), \quad \text{and} \\
(M \bullet K)^\circ &= (M)^\circ \bullet (K)^\circ,
\end{aligned}$$

where V and W are values, \mathcal{M} and \mathcal{N} are not values, and x, y, α, β are fresh.

We need the redundant definition of $(\langle \mathcal{M}, \mathcal{N} \rangle)^\circ$ for a technical reason, and it is necessary in order to show Proposition 7.25.

The transformation $(-)^{\circ}$ preserves typing.

Proposition 7.22. Let M be a term, K be a cotermin, and S be a statement of $\text{DC}\mu\nu$. The following claims hold.

- (1) If $\Gamma \vdash_{\text{DC}\mu\nu} \Delta \mid M : A$ is provable, then $\Gamma \vdash_{\text{DC}\mu\nu} \Delta \mid (M)^\circ : A$ holds.
- (2) If $K : A \mid \Gamma \vdash_{\text{DC}\mu\nu} \Delta$ is provable, then $(K)^\circ : A \mid \Gamma \vdash_{\text{DC}\mu\nu} \Delta$ holds.
- (3) If $\Gamma \mid S \vdash_{\text{DC}\mu\nu} \Delta$ is provable, then $\Gamma \mid (S)^\circ \vdash_{\text{DC}\mu\nu} \Delta$ holds.

Proof. They are shown simultaneously by induction on M, K , and S . □

The transformation $(-)^{\circ}$ satisfies the following basic properties.

Lemma 7.23. Let V be a value, M and N be terms, D be an expression of $\text{DC}\mu\nu$. Then the following claims hold.

- (1) M is a value iff $(M)^\circ$ is a value.
- (2a) $\langle (M)^\circ, (N)^\circ \rangle \xrightarrow{*}_{\text{wCBV}} \langle \langle M, N \rangle \rangle^\circ$.
- (2b) $\langle (M)^\circ \rangle \text{inl} \xrightarrow{*}_{\text{wCBV}} \langle \langle M \rangle \text{inl} \rangle^\circ$, and $\langle (M)^\circ \rangle \text{inr} \xrightarrow{*}_{\text{wCBV}} \langle \langle M \rangle \text{inr} \rangle^\circ$.
- (2c) $\text{in}^{\mu X.A} \langle (M)^\circ \rangle \xrightarrow{*}_{\text{wCBV}} (\text{in}^{\mu X.A} \langle M \rangle)^\circ$.

- (2d) $\text{coitr}_z^A \langle (M)^\otimes, (N)^\otimes \rangle \longrightarrow_{\text{wCBV}} (\text{coitr}_z^A \langle M, N \rangle)^\otimes$.
 (3a) $\text{mono}_{B,C,x}^{X,A} \{(M)^\otimes, (N)^\otimes\} \longrightarrow_{\text{wCBV}}^* (\text{mono}_{B,C,x}^{X,A} \{M, N\})^\otimes$.
 (3b) $\text{mono}_{B,C,\alpha}^{X,A} \{(K)^\otimes, (L)^\otimes\} \longrightarrow_{\text{wCBV}}^* (\text{mono}_{B,C,\alpha}^{X,A} \{K, L\})^\otimes$.
 (4) $D \longrightarrow_{\text{CBV}}^* (D)^\otimes$.

Proof. The claim (1) is shown by the definition of $(-)^\otimes$. The claims (2a), (2b), (2c), and (2d) are shown by (1) and ζ -rules. The claims (3a) and (3b) are shown by induction on $\|C\|_X$ using (2a), (2b), (2c), and (2d). The claim (4) is shown by induction on D . \square

The transformation $(-)^\otimes$ preserves substitution of a value for a variable, and of a coterm for a covariable.

Lemma 7.24. $(D[V/x])^\otimes = (D)^\otimes[(V)^\otimes/x]$ and $(D[K/\alpha])^\otimes = (D)^\otimes[(K)^\otimes/\alpha]$.

Proof. The former claim is shown by induction on D using Lemma 7.23 (1). The latter one is shown by induction on D . \square

The transformation $(-)^\otimes$ translates one step reduction of $\longrightarrow_{\text{CBV}}$ into zero or more steps reduction of $\longrightarrow_{\text{wCBV}}$.

Proposition 7.25. $D \longrightarrow_{\text{CBV}} E$ implies $(D)^\otimes \longrightarrow_{\text{wCBV}}^* (E)^\otimes$. In particular, if $D \longrightarrow_{\text{CBV}} E$ by $(\beta)_v$ or $(\eta)_v$, then $(D)^\otimes \longrightarrow_{\text{wCBV}}^+ (E)^\otimes$ holds.

Proof. The claim is shown by induction on the definition of $\longrightarrow_{\text{CBV}}$. We show the cases of $(\beta L)_v$, $(\beta\mu)_v$, and $(\zeta\wedge_1)_v$.

The case of $(\beta L)_v$ is proved by Lemma 7.24. We have $(V \bullet x.(S))^\otimes = (V)^\otimes \bullet x.(S)^\otimes \longrightarrow_{\text{wCBV}} (S)^\otimes[(V)^\otimes/x]$. By Lemma 7.24, the last statement is $(S[V/x])^\otimes$.

The case of $(\beta\mu)_v$ is proved by Lemma 7.23 (3b). We have $(\text{in}^{\mu X.C} \langle V \rangle \bullet \text{itr}_\alpha^A [K, L])^\otimes = \text{in}^{\mu X.C} \langle (V)^\otimes \rangle \bullet \text{itr}_\alpha^A [(K)^\otimes, (L)^\otimes] \longrightarrow_{\text{wCBV}} ((V)^\otimes \bullet \text{mono}_{\mu X.C, A, \beta}^{X,C} \{ \text{itr}_\alpha^A [(K)^\otimes, \beta], (K)^\otimes \}) \cdot \alpha \bullet (L)^\otimes = ((V)^\otimes \bullet \text{mono}_{\mu X.C, A, \beta}^{X,C} \{ (\text{itr}_\alpha^A [K, \beta])^\otimes, (K)^\otimes \}) \cdot \alpha \bullet (L)^\otimes$. By Lemma 7.23 (3b), the last statement is reduced to $((V)^\otimes \bullet (\text{mono}_{\mu X.C, A, \beta}^{X,C} \{ \text{itr}_\alpha^A [K, \beta], K \})^\otimes) \cdot \alpha \bullet (L)^\otimes$ by $\longrightarrow_{\text{wCBV}}^*$. Therefore this statement is $((V \bullet \text{mono}_{\mu X.C, A, \beta}^{X,C} \{ \text{itr}_\alpha^A [K, \beta], K \}) \cdot \alpha \bullet L)^\otimes$.

The case of $(\zeta\wedge_1)_v$ is proved by the definition of $(-)^\otimes$. We consider the subcase of $\langle M, N \rangle \longrightarrow_{\text{CBV}} (M \bullet x.(\langle x, N \rangle \bullet \alpha)) \cdot \alpha$, where M and N are not values. Hence we have $(\langle M, N \rangle)^\otimes = ((M)^\otimes \bullet x.((N)^\otimes \bullet y.(\langle x, y \rangle \bullet \beta)).\beta \bullet \alpha) \cdot \alpha = ((M)^\otimes \bullet x.((\langle x, N \rangle)^\otimes \bullet \alpha)) \cdot \alpha = (M \bullet x.(\langle x, N \rangle \bullet \alpha))^\otimes \cdot \alpha$. The other subcase of $(\zeta\wedge_1)_v$ for $\langle N, V \rangle$ with a non-value N is shown in the similar way.

The other cases are also shown by the induction hypothesis. \square

Then we can show the Church-Rosser property of $\longrightarrow_{\text{CBV}}$ and $\longrightarrow_{\text{CBN}}$.

Theorem 7.26. The reduction relations $\longrightarrow_{\text{CBV}}$ of CBV $\text{DC}\mu\nu$ and $\longrightarrow_{\text{CBN}}$ of CBN $\text{DC}\mu\nu$ satisfy the Church-Rosser property.

Proof. We first show the Church-Rosser property of $\longrightarrow_{\text{CBV}}$.

Assume that $D \longrightarrow_{\text{CBV}}^* D'$ and $D \longrightarrow_{\text{CBV}}^* D''$ hold. By Proposition 7.25, we have $(D)^\otimes \longrightarrow_{\text{wCBV}}^* (D')^\otimes$ and $(D)^\otimes \longrightarrow_{\text{wCBV}}^* (D'')^\otimes$. By the Church-Rosser property of $\longrightarrow_{\text{wCBV}}$, there exists E such that $(D')^\otimes \longrightarrow_{\text{wCBV}}^* E$ and $(D'')^\otimes \longrightarrow_{\text{wCBV}}^* E$. Therefore, by Lemma 7.19 (1) and Lemma 7.23 (4), we have $D' \longrightarrow_{\text{CBV}}^* (D')^\otimes \longrightarrow_{\text{CBV}}^* E$ and $D'' \longrightarrow_{\text{CBV}}^* (D'')^\otimes \longrightarrow_{\text{CBV}}^* E$.

The Church-Rosser property of $\longrightarrow_{\text{CBN}}$ is shown by the former result and the duality between $\longrightarrow_{\text{CBV}}$ and $\longrightarrow_{\text{CBN}}$ (Prop 7.20). \square

We will prove strong normalization of CBV $\text{DC}\mu\nu$ and CBN $\text{DC}\mu\nu$. This property is shown by using the strong normalization result of the weak call-by-value and the weak call-by-name $\text{DC}\mu\nu$ (Proposition 7.11).

We define the following rank of expressions in $\text{DC}\mu\nu$. This rank is used to show that there is no infinite sequence of ζ -rules.

Definition 7.27. Let D be an expression in $\text{DC}\mu\nu$. The rank $r(D)$ of D is defined by:

$$\begin{aligned} r(x) &= r(\alpha) = 0, \\ r([K]\text{not}) &= r(\text{fst}[K]) = r(\text{snd}[K]) = r(\text{out}^{\nu X.A}[K]) = r(K), \\ r([K, L]) &= r(\text{itr}_\alpha^A[K, L]) = r(K) + r(L), \\ r(\text{not}\langle M \rangle) &= r(M), \\ r(\langle M, N \rangle) &= r(M) + r(N) + 2, \\ r(\langle M, V \rangle) &= r(\langle V, M \rangle) = r(M) + r(V) + 1, \\ r(\langle V, W \rangle) &= r(V) + r(W), \\ r(\langle M \rangle \text{inl}) &= r(\langle M \rangle \text{inr}) = r(\text{in}^{\mu X.A}\langle M \rangle) = r(M) + 1, \\ r(\langle V \rangle \text{inl}) &= r(\langle V \rangle \text{inr}) = r(\text{in}^{\mu X.A}\langle V \rangle) = r(V), \\ r(\text{coitr}_x^A\langle M, N \rangle) &= r(M) + r(N) + 1, \\ r(\text{coitr}_x^A\langle M, V \rangle) &= r(M) + r(V), \\ r(x.(S)) &= r((S).\alpha) = r(S), \quad \text{and} \\ r(M \bullet K) &= r(M) + r(K), \end{aligned}$$

where V and W are values, and M and N are not values.

The rank $r(D)$ counts the number of redexes of $(\zeta \wedge_1)_v$, $(\zeta \wedge_2)_v$, $(\zeta \vee_1)_v$, $(\zeta \vee_2)_v$, $(\zeta \mu)_v$, and $(\zeta \nu)_v$ -rules. We write $D \longrightarrow_{\zeta_v} E$ when D is reduced to E by one step $(\zeta)_v$ -reduction.

Lemma 7.28. Let D and E be expressions of $\text{DC}\mu\nu$. Then, the following claims hold.

- (1) If $D \longrightarrow_{\zeta_v} E$, then $r(D) > r(E)$.
- (2) There is no infinite sequence of $(\zeta)_v$ -reduction.

Proof. The claim (1) is shown by induction on D . The claim (2) is shown by (1). \square

We then show strong normalization of CBV $\text{DC}\mu\nu$ and CBN $\text{DC}\mu\nu$.

Theorem 7.29 (Strong normalization of CBV $\text{DC}\mu\nu$ and CBN $\text{DC}\mu\nu$). *The following claims hold.*

- (1) Every typable expression is strongly normalizing in CBV $\text{DC}\mu\nu$.
- (2) Every typable expression is strongly normalizing in CBN $\text{DC}\mu\nu$.

Proof. We first show the call-by-value case. Assume that D is typable in $\text{DC}\mu\nu$ and there is an infinite reduction sequence

$$D \longrightarrow_{\text{CBV}} D_1 \longrightarrow_{\text{CBV}} \dots$$

starting from D . Then $(D)^\circledast$ is typable by Proposition 7.22, and we have

$$(D)^\circledast \xrightarrow{*}_{\text{wCBV}} (D_1)^\circledast \xrightarrow{*}_{\text{wCBV}} \dots$$

by Proposition 7.25. From the strong normalization result of the weak call-by-value $\text{DC}\mu\nu$ (Proposition 7.11), there is some D_k such that

$$(D_k)^\circledast = (D_{k+1})^\circledast = \dots$$

By the latter part of Proposition 7.25, we have the following infinite sequence of $(\zeta)_v$ -reduction:

$$D_k \longrightarrow_{\zeta_v} D_{k+1} \longrightarrow_{\zeta_v} \dots$$

This contradicts Lemma 7.28 (2).

The call-by-name case is proved by strong normalization of $CBV\ DC_{\mu\nu}$ and the duality between $CBV\ DC_{\mu\nu}$ and $CBN\ DC_{\mu\nu}$ (Proposition 7.20). \square

8. CONCLUSION

We have introduced the non-deterministic system $DC_{\mu\nu}$ by extending the dual calculus given in [27] with inductive types and coinductive types. Besides the same duality of the original dual calculus, we have shown the duality of inductive and coinductive types, by giving the involution that maps terms and coterms for inductive types to coterms and terms of coinductive types respectively and vice versa, and maps their reduction rules to each other. We have proved its strong normalization by translating it into the second-order dual calculus $DC2$.

The second-order dual calculus $DC2$ also have been introduced. Its strong normalization have been shown by translating it into the second-order symmetric lambda calculus.

We have finally introduced the call-by-value system $CBV\ DC_{\mu\nu}$ and the call-by-name system $CBN\ DC_{\mu\nu}$ of the dual calculus with inductive and coinductive types. We have shown the duality of call-by-value and call-by-name with inductive and coinductive types, their Church-Rosser property, and their strong normalization. Their strong normalization have been shown by translating them into $DC_{\mu\nu}$.

The first author introduced the call-by-value and call-by-name dual calculi with recursive types [11, section 4.2]. In these systems, a recursive type $\text{rec } X.A$ can be defined for any type A . If we assume that $\text{rec } X.A$ can be defined only if every X positively occurs in A , then we can define two provability-preserving transformations from the dual calculi with recursive types into $DC_{\mu\nu}$. The one translates a recursive type to an inductive type, and the other translates a recursive type to a coinductive type. We could not straightforwardly show that these transformations preserve reductions (or equations) since some additional rules such as η -rules for connectives seem to be required. This problem would be future work.

The duality of call-by-value and call-by-name in $\lambda\mu$ -calculus is shown by using the dual calculi in [28]. Since our systems $CBV\ DC_{\mu\nu}$ and $CBN\ DC_{\mu\nu}$ are extensions of his dual calculi, we could show the duality of call-by-value and call-by-name in $\lambda\mu$ -calculus with inductive and coinductive definitions, by using our systems $CBV\ DC_{\mu\nu}$ and $CBN\ DC_{\mu\nu}$. It would be future work.

A reduction-based duality between call-by-value and call-by-name in the $\lambda\mu$ -calculi was presented in [12], by refining Wadler's result [28]. Extending the result given in [12] with inductive and coinductive types would be future work.

Our systems use the iteration for inductive types. An extension of the iteration to primitive recursion would be future work.

A CPS translation from the dual calculus to λ -calculus was given in [27]. Extending this CPS translation to the systems with inductive and coinductive types would be future work.

ACKNOWLEDGMENT

We would like to thank Professor Philip Wadler for discussions and suggestions. We would also like to thank Dr. Alwen Tiu, and Professor Dieter Spreen for discussions. We would also like to thank anonymous referees for valuable comments.

REFERENCES

- [1] D. Baelde. Least and greatest fixed points in linear logic. *ACM Transactions on Computational Logic*, 13 (1): Article 2, 2012.
- [2] F. Barbanera and S. Berardi. A symmetric lambda calculus for classical program extraction. *Information and Computation*, 125 (2): 103–117, 1996.
- [3] W. Buchholz, S. Feferman, W. Pohlers, and W. Sieg. Iterated inductive definitions and subsystems of analysis: Recent proof-theoretical studies. *Lecture Notes in Mathematics*, 897, Springer, 1981.
- [4] P.-L. Curien and H. Herbelin. The duality of computation. In *Proceedings of the 5th ACM SIGPLAN International Conference on Functional Programming (ICFP)*, 233–243, 2000.
- [5] A. Filinski. Declarative continuations and categorical duality. *Master’s thesis*, Computer Science Department, University of Copenhagen, DIKU Report 89/11, 1989.
- [6] H. Geuvers. Inductive and coinductive types with iteration and recursion. In *Proceedings of the 1992 workshop on Types for Proofs and Programs (TYPES)*, 183–207, 1992.
- [7] J.-Y. Girard. *Proof theory and logical complexity*, Bibliopolis, 1987.
- [8] T.G. Griffin. A formulae-as-types notion of control. In *Proceedings of the 17th ACM SIGPLAN-SIGACT symposium on Principles of programming languages*, 47–58, 1990.
- [9] H. Herbelin. C’est maintenant qu’on calcule au cœur de la dualité Habilitation à diriger les recherches, L’Université Paris 11, 2005.
- [10] Y. Kakutani. Duality between call-by-name recursion and call-by-value iteration. In *Proceedings of the 16th International Workshop on Computer Science Logic, CSL*, Lecture Notes in Computer Science 2471: 506–521, 2002.
- [11] D. Kimura. Call-by-value is dual to call-by-name, extended. In *Proceedings of Programming Languages and Systems, 5th Asian Symposium (APLAS)*, Lecture Notes in Computer Science, 4807: 415–430, 2007.
- [12] D. Kimura. Duality between call-by-value reductions and call-by-name reductions. *IPSJ Journal*, 48(4): 1721–1757, 2007.
- [13] D. Kimura and M. Tatsuta. Dual calculus with inductive and coinductive types. In *Proceedings of 20th International Conference on Rewriting Techniques and Applications (RTA)*, Lecture Notes in Computer Science, 5595: 224–238, 2009.
- [14] R. McDowell and D. Miller. Cut-elimination for a logic with definitions and induction. *Theoretical Computer Science*, 232 (1-2): 91–119, 2000.
- [15] N. P. Mendler. Inductive types and type constraints in the second-order lambda calculus. *Annals of Pure and Applied Logic*, 51 (1-2): 159–172, 1991.
- [16] A. Momigliano and A. Tiu. Induction and co-induction in sequent calculus. *Types for Proofs and Programs International Workshop (TYPES), Revised Selected Papers*, Lecture Notes in Computer Science, 3085: 293–308, 2004.
- [17] A. Tiu and Momigliano. Cut elimination for a logic with induction and co-induction. manuscript, available at arxiv.org, 2010.
- [18] Nordström, B., Petersson, K. and Smith, J.M. *Programming in Martin-Löf’s type theory*. Oxford University Press, 1990.
- [19] M. Parigot. $\lambda\mu$ -calculus: an algorithmic interpretation of classical natural deduction. In *Proceedings of International Conference on Logic Programming and Automated Deduction (LPAR)*, Lecture Notes in Computer Science, 624: 190–201, 1992.
- [20] M. Parigot. Strong normalization for second order classical natural deduction. In *Journal of Symbolic Logic*, 62(4): 1461–1479, 1997.
- [21] M. Parigot. Strong normalization of second order symmetric lambda-calculus. In *Proceedings of Foundations of Software Technology and Theoretical Computer Science*, Lecture Notes in Computer Science, 1974: 442–453, 2000.
- [22] C. Paulin-Mohring. Inductive definitions in the system Coq — Rules and properties. In *Proceedings of Typed Lambda Calculi and Applications (TLCA)*, Lecture Notes in Computer Science, 664: 328–345, 1993.
- [23] P. Selinger. Control categories and duality: on the categorical semantics of the lambda-mu calculus. *Mathematical Structures in Computer Science*, 207–260, 2001.
- [24] M. Tatsuta. Realizability interpretation of coinductive definitions and program synthesis with streams. *Theoretical Computer Science*, 122(1–2): 119–136, 1994.

- [25] M. Tatsuta. Simple saturated sets for disjunction and second-order existential quantification. In *Proceedings of 8th International Conference on Typed Lambda Calculi and Applications (TLCA)*, Lecture Notes in Computer Science, 4583: 366–380, 2007.
- [26] N. Tzevelekos. Investigations on the dual calculus. *Theoretical Computer Science*, 360: 289–326, 2006.
- [27] P. Wadler. Call-by-value is dual to call-by-name. In *Proceedings of International Conference on Functional Programming (ICFP)*, 189–201, 2003.
- [28] P. Wadler. Call-by-value is dual to call-by-name, reloaded, In *Proceedings of Rewriting Techniques and Applications (RTA)*, Lecture Notes in Computer Science, 3467: 185–203, 2005.